



Bootstrapped Learning (BL)

Proposer Information Pamphlet (PIP)

for

Broad Agency Announcement 07-04

**BAA Bootstrapped Learning (BL), SOL BAA 07-04, Proposals Due: Initial Closing:
18 January 2007, Final Closing: 14 November 2007, POC: Dr. Dan Oblinger,
DARPA/IPTO; FAX: (703) 741-7804**

Defense Advanced Research Projects Agency
Information Processing Technology Office
3701 North Fairfax Drive
Arlington, VA 22203-1714

TABLE OF CONTENTS

CHAPTER 1. PROGRAM OBJECTIVES AND DESCRIPTION	5
WHAT IS BOOTSTRAPPED LEARNING?	5
WHAT IS THE BOOTSTRAPPED LEARNING PROGRAM?	6
WHAT ARE THE BLP OBJECTIVES?	6
THE BL FRAMEWORK	8
<i>The Learning System and its Environment.....</i>	<i>8</i>
<i>The Elaborated Framework</i>	<i>9</i>
<i>Elaboration of the Learning System.....</i>	<i>10</i>
<i>Elaboration of the Interaction Specification.....</i>	<i>14</i>
<i>Elaboration of the Curriculum.....</i>	<i>16</i>
<i>Procedural Overview</i>	<i>17</i>
CHAPTER 2. PROGRAMMATIC DETAILS	19
PHASING	19
ORGANIZATION.....	20
<i>Learning System Team.....</i>	<i>20</i>
<i>Curriculum Team</i>	<i>21</i>
AWARDS	21
CHAPTER 3. PERFORMER REQUIREMENTS	23
INTRODUCTION	23
REQUIREMENTS OF THE LEARNING TEAM	23
<i>Integration Architecture.....</i>	<i>23</i>
<i>Learning Processes.....</i>	<i>24</i>
<i>Desired Characteristics of a Learning System Developer</i>	<i>24</i>
<i>Key Technical Questions to Be Addressed by a Learning Team Bidder.....</i>	<i>24</i>
REQUIREMENTS OF THE CURRICULUM TEAM	26
<i>Genetic Knowledge</i>	<i>26</i>
<i>Bootstrapped Learning Support Infrastructure</i>	<i>27</i>
<i>Curriculum Ladders.....</i>	<i>28</i>
<i>The Electronic Instructor</i>	<i>30</i>
<i>Injected Knowledge.....</i>	<i>32</i>
<i>Formal Evaluations.....</i>	<i>32</i>
<i>Mitigation Strategy for Potential Conflict of Interest</i>	<i>32</i>
<i>Desired Characteristics of a Curriculum Developer</i>	<i>33</i>
<i>Key Technical Questions to Be Addressed by a Curriculum Developer Bidder</i>	<i>34</i>
FORMAL EVALUATIONS OF THE LEARNING SYSTEMS	34
<i>How the Claims Will Be Tested.....</i>	<i>34</i>
<i>The Assessment Protocol.....</i>	<i>35</i>
<i>Go/No-Go Tests.....</i>	<i>36</i>
<i>Additional Tests.....</i>	<i>37</i>
<i>Details of Human Testing.....</i>	<i>37</i>
CHAPTER 4. GENERAL INFORMATION	38
WEB SITE, ONGOING Q&A AND OTHER EXTERNAL INFORMATION	38
OFFEROR ELIGIBILITY	38
ADMINISTRATIVE NOTES	38
HUMAN USE	39
SECURITY CLASSIFICATION	39
PUBLICATION APPROVAL	39
EXPORT LICENSES	40

CHAPTER 5. SUBMISSION PROCESS	41
CHAPTER 6. REPORTING REQUIREMENTS/PROCEDURES:	42
CHAPTER 7. PROPOSAL PREPARATION AND FORMAT	43
SECTION I. ADMINISTRATIVE	43
<i>Confirmation Sheet</i>	43
SECTION II. DETAILED PROPOSAL INFORMATION.....	43
SECTION III. ADDITIONAL TECHNICAL INFORMATION	54
SECTION IV. COST PROPOSAL	55
<i>A. Cover sheet</i>	55
<i>B. Detailed cost breakdown</i>	55
<i>C. Government Furnished Property</i>	57
CHAPTER 8. EVALUATION AND FUNDING PROCESSES	58
CHAPTER 9. BAA CORRESPONDENCE AND ADMINISTRATIVE ADDRESSES	61
APPENDIX A: EXTERNAL MATERIALS	62
USE OF MATERIALS	62
LANGUAGES	62
APPENDIX B: BACKGROUND & MOTIVATION	63
CONTRASTING BL WITH TRADITIONAL ML	63
WHY IS THERE AN OPPORTUNITY FOR BL NOW?	66
WHY IS THERE CONFIDENCE THAT THE BL CONCEPT IS FEASIBLE?	66
BL WILL IMPACT MULTIPLE RESEARCH COMMUNITIES	67
BL WILL SIGNIFICANTLY EXPAND THE USE OF COMPUTERS.....	68
<i>Encapsulated Trainability</i>	68
<i>Field Trainable Department of Defense (DoD) Systems</i>	68
<i>Commercial and Other Applications</i>	69
APPENDIX C: DETAILS OF THE TECHNICAL APPROACH	70
INITIAL ELEMENTS OF THE PROGRAM	70
<i>Bootstrapped Learning Components and Interlingua</i>	70
<i>Interaction Language</i>	71
<i>Natural Instruction Methods</i>	72
<i>Ancillary Competencies</i>	77
<i>Curricula and Curriculum Ladders</i>	78
<i>Learning Processes</i>	79

TABLE OF FIGURES

Figure 1: The Basic Framework	8
Figure 2: The Elaborated Framework	10
Figure 3: The Structure of a BLC	13
Figure 4: Bootstrapped Learning Schedule	19
Figure 5: Organization and Structure of Program	20
Figure 6: The BL Metrics	36
Figure 7: Natural Instruction Makes BL Feasible	67
Figure 8: Taxonomy of Natural Instruction Methods	74

TABLE OF TABLES

Table 1: Go/No-Go Thresholds for Phases 2 and 3	37
Table 2: Comparison of Bootstrapped Learning and Traditional Machine Learning	63

CHAPTER 1. PROGRAM OBJECTIVES AND DESCRIPTION

What Is Bootstrapped Learning?

Bootstrapped Learning (BL) is autonomous, domain-independent learning that builds upon the knowledge gained from prior learning via interactions with an instructor who is assumed to possess the competencies that are to be learned. “Bootstrapped” refers to the notion that learning takes place in the form of a series of instructional experiences (lessons), in which each builds upon the knowledge gained from previous lessons.

There are two primary defining characteristics of a system that carries out BL (a “BL system” or “learning system”):

- A BL system is composed of multiple learning processes that are completely *independent* of any given task to be learned. In particular, they are not specialized to a specific task domain and have no externally settable configuration parameters. Instead, these learning processes are specialized to types of instruction. For example, one learning process might be geared toward making inferences from demonstrated examples of a script to be learned. However, that same learning process would apply equally to learning how to bake a cake or learning how to tie a shoe. The key distinction here is that the learning processes are specialized to methods of instruction, not to task domains. The only stipulation for this particular learning process is that tasks to be learned must fall into the very broad category of tasks that can be expressed as processes.
- The configuration (i.e., bias) of a learning process must be bootstrapped entirely from prior learning. For example, a model of the effects of a set of individual actions on the world might be bootstrapped from prior lessons. That knowledge might then be used as a starting point for a learning process that produces a script from demonstrations of that script, where each element in the script is one of the previously learned actions. Indeed, the learning algorithms of today employ a great variety of bias knowledge in order to produce effective learning performance. In bootstrapped learning, all of that bias knowledge must be expressible in a format that allows it, in principle, to be learnable from prior experience. It is this cyclical nature of bootstrapped learning that allows bootstrapping to occur.

Thus the general applicability of a successful BL system is assured by the fact that its learning processes are developed and configured independently of any particular task domain.

The overall charge of the program is the exploration of very general systems that can be “guided” by an instructor to learn any performance task that is expressible using some pre-enumerated set of teaching methods. *Such trainable systems would enable an entirely new computing usage model, in which the computing system behaviors are taught rather than programmed.*

BL constitutes a departure from the objectives of traditional machine learning (ML) research. In particular, a central assumption in BL is that there is an expert instructor (either human or humanlike) who possesses knowledge and capabilities that the student is expected to learn.

By contrast, traditional ML focuses primarily on the task of acquiring knowledge that is not possessed by the ML system's user. In some ways ML can be seen as the problem of knowledge *discovery* and BL as the problem of knowledge *communication* from instructor to student. Although BL is not ML, components of BL will likely benefit from ML research.

There are many ways that human instructors interact with students to effect learning. These range from formal pedagogic techniques to informal modes of communication such as gesturing. While the techniques used in traditional educational settings are the most commonly known, BL is not limited to the types of instruction that take place in a classroom setting; rather, it seeks to include the most ubiquitous and useful of the myriad disparate ways that knowledge can be transferred between humans. These knowledge transfer techniques will be termed *natural instruction (NI) methods*.

Natural instruction (NI) methods - specific ways that human instructors interact with their students; each NI method can utilize one or more *interaction modalities*, which are abstractions of the physical communication channels that embody the instruction (e.g., speech, gesture, photograph, sketch, etc.).

One common class of NI methods is *instruction by examples*. An instructor might provide a small set of examples (e.g., photographs) that illustrate a concept to be learned, possibly commenting on important features of the examples using (controlled) speech while also using hand gestures to point at salient features of the examples. Alternatively, the teacher might demonstrate a procedure to be learned, such as when a swimming instructor demonstrates a stroke. Another common class of NI methods is *instruction using feedback*, which includes activities such as having the student refine skills, often independently, within a practice context that includes constraints and a goal.

What Is the Bootstrapped Learning Program?

The Bootstrapped Learning Program (BLP) is a new research effort by the Information Processing Technology Office (IPTO) of the Defense Advanced Research Projects Agency (DARPA) to create and demonstrate one or more complete BL systems. This involves producing "electronic students" (i.e., the BL systems), electronic curricula, and a natural instruction methods framework that allows the curricula to be taught to the students. The program will also support the development of an "automated tutor" that will be used to teach and evaluate the learning systems.

What Are the BLP Objectives?

The principal objective of the BLP is to show the feasibility of the BL concept and thereby launch a new ML community that emphasizes a new class of ML algorithms which are tools for pedagogical communication. Toward that end, the initial goal is to develop prototype learning systems that actually demonstrate BL. With the realization of such systems, it is then possible to conduct formal evaluations to test the following two scientific claims:

- Claim #1: *domain independence*. A BL-based system is capable of learning a wide range of performance tasks in a wide range of problem domains, based on abstracted NI methods, with *zero* reprogramming/reconfiguration of the learning system between learning sessions.
- Claim #2: *good performance*. The BL accomplished by such a system compares “reasonably well” with learning by a human, when each is given the “same” background knowledge followed by the “same” instructional content using the same NI methods. (All senses of “same” are made precise by this program, as is the definition of “reasonably well.”)

In order to achieve these principal objectives, the BLP will conduct a number of activities:

- Create a variety of *domain-independent* learning processes that can learn from an instructor who is teaching with each of the NI methods;
- Create an integrated learning system consisting of multiple learning processes and a controller for how and when to apply them;
- Create a curriculum system that transforms instructional materials into a form suitable for instruction via NI methods
- Create multiple curricula that utilize the NI methods to teach a single subject;
- Demonstrate that a single learning system can learn from multiple curricula without any reconfiguration between domains; and
- Demonstrate that a learning system can learn reasonably well when compared to humans who learn from the “same” curriculum. (See Chapter 3 for a discussion of this comparison)

In the course of substantiating the two claims above, the BLP is expected to produce tangible deliverables, in four areas:

- One or more “electronic student” system(s) that have a number of learning processes which are highly reusable because they were designed to be domain-independent; that embody much stronger forms of learning, driven by instruction, than the ML techniques of today; and whose performance compares to human learning performance.
- A diverse collection of curricula expressed in a standardized format that will foster a new “instruction-based learning” research community.
- A test harness that, for the first time, will allow *individual* researchers to download entire BL systems and curricula so they can develop and test empirically new NI-based BL processes in the context of a complete, “closed” electronic-student / electronic-instructor system.
- The core technology for a new class of *field-trainable* military computing systems.

The BL Framework

This section lays out the BL framework by providing an overview of the framework components and their interactions. In addition, this section explains key concepts and defines terminology that will be revisited throughout this document.

The Learning System and its Environment

Bootstrapped learning occurs as the result of interactions between an electronic student or *learning system* and a learning context for a specific task, referred to as a *curriculum*. Interactions among these entities are governed by a fixed API (application programmers interface), called the *interaction language API*. Figure 1 depicts these three components, which comprise the basic BL framework.

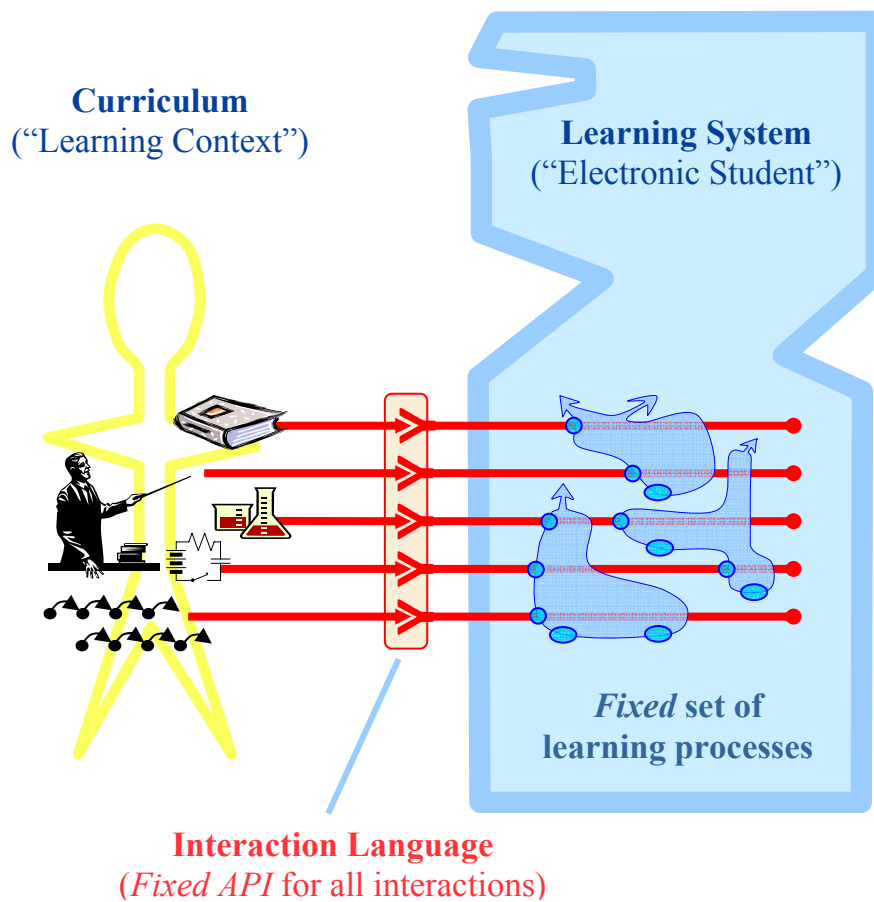


Figure 1: The Basic Framework

Learning system - an automated, instructable system capable of acquiring new competencies across task domains without modification; consists of a set of learning processes and a top-level controller that evaluates, processes, and responds to inputs.

Interaction language - a communication specification tailored to provide a consistent representation of instructional information that supports a wide class of instructional methods; the language in which all curriculum lessons are encoded. All interaction between the learning system and its environment is encoded in the interaction language, including interaction with the instructor and interaction with the world.

Curriculum (also "Laddered Curriculum", or simply "Ladder") - a collection of information and processes that, in its entirety, provides an environment sufficient for a learning system to learn a specific goal behavior; includes common knowledge, domain-specific injected knowledge, lessons, automated instruction, a world simulator, and a simulator interface (these terms are explained below). In short, a curriculum is everything outside of the learning system that is needed for learning a single competency.

Domain-independent instruct-able machine learning requires significant advancement beyond existing technology. In order to focus research on the most fundamental and groundbreaking aspects of the problem, the BL program explicitly precludes work on the automation of abstracting perceptual interaction modalities. It is assumed that other efforts at DARPA and elsewhere will address the issue of automatically obtaining a high-level representation of the instructor's physical actions (e.g., gestures and utterances), as well as world state change. This program uses the interaction language API as a means of transcending those issues. Essentially, this fixed API allows the learning system to interact with the electronic instructor and world using an abstracted representation. Thus, to ensure real-world applicability, it is important to implement a high-level representation that would be consistent with the output of an automated abstraction process that operates directly on raw perceptual input from the instructor and the world.

The Elaborated Framework

Due to the considerable scope of the BLP, an atypical level of specificity is required for the framework definition in order to render the program goals tractable. At such a level of detail, this framework allows performers to focus on specific, well-defined classes of research issues related to bootstrapped learning. These issues will surface in the course of elucidating the framework.

The purpose of this section is to flesh out the three main components of the BL framework: the learning system, the interaction language, and the learning context, i.e., the curriculum. These components are depicted in Figure 2. The blue rectangle represents the learning system, the yellow ladder represents the curriculum ladder, and the pink bar separating them

corresponds to the interaction language API, which provides a consistent interface for their interaction.

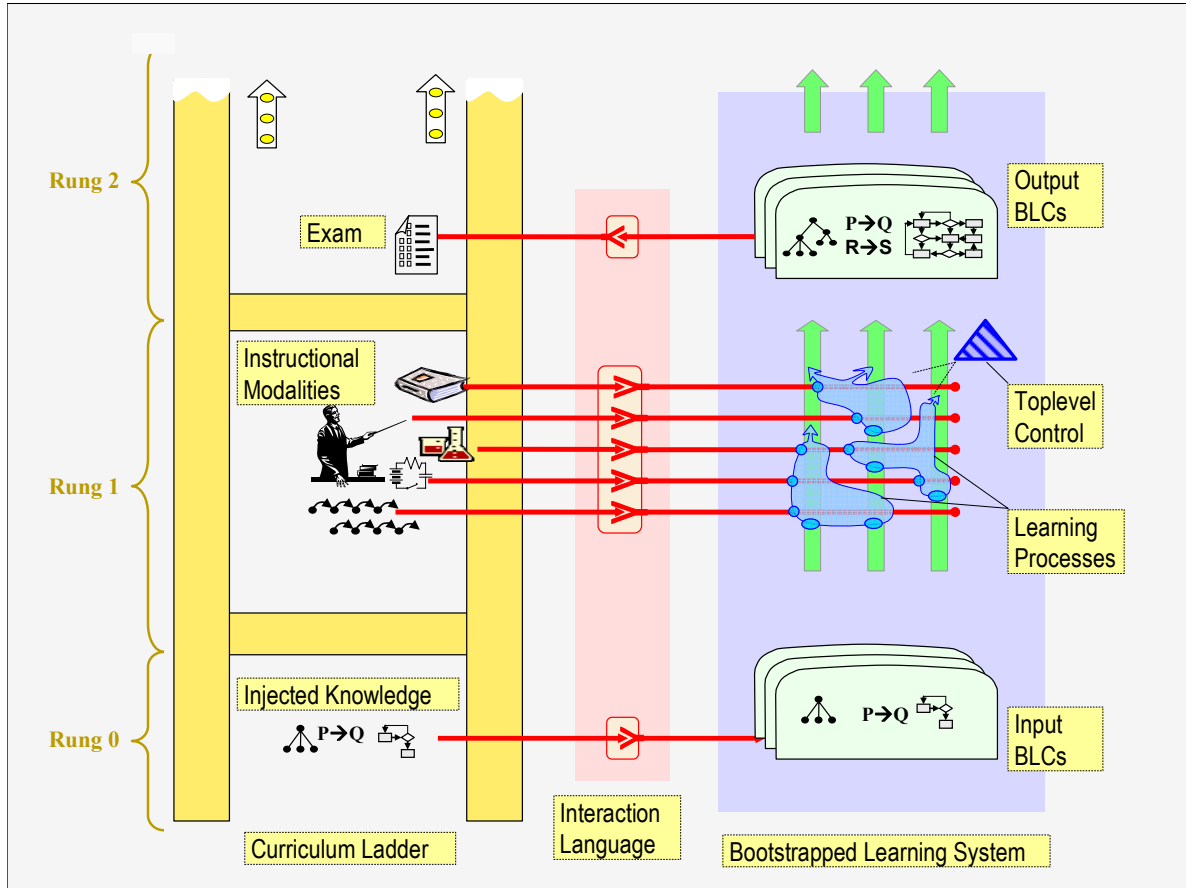


Figure 2: The Elaborated Framework

Elaboration of the Learning System

A learning system consists of a set of *learning processes*, an *integration architecture* (depicted as the “Top-level Control”), and (during execution) bootstrapped learning components (BLCs) containing injected and learned knowledge. BLC is described below in the section on “The Bootstrapping Process.”

Learning process - a domain-independent process that accepts prior knowledge BLCs along with NI method interaction with an instructor and the world; produces new or refined BLCs that can serve as the starting point for learning by other learning processes.

Integration architecture - top-level controller for the learning system responsible for all reasoning required to control the learning processes in order to achieve bootstrapped learning; evaluates, processes, and responds to interaction language communiqués in the context of existing encoded knowledge (in the input BLCs) and through the selective utilization of appropriate learning processes.

The integration architecture receives input from the curriculum in the form of NI method instructions coded in the interaction language. This input is considered in the context of acquired knowledge as well as meta-knowledge about the goal state, and subsequent learning steps. Ultimately, a decision is made about which learning process to recruit in order to encode the learning input. As will be conveyed explicitly in the next section, the manner in which learned knowledge is encoded is the key to bootstrapping.

Knowledge Representation

A departure of BL from traditional ML is the insistence that **all** configuration and background knowledge about how to learn is derived from prior learning. This radical edict implies that the **output** language of a learning system used to represent knowledge must map onto the **input** language of a learning system, including provisions for the representation of the entire learning *bias* (e.g., configuration, background knowledge, etc.). Furthermore, a necessary condition for indefinite bootstrapping is that each aspect of the language required for configuration be generate-able by at least one BL process. Therefore, bootstrapped learning will only be possible if:

- The knowledge representation language is flexible enough to be able to encode all knowledge needed for strong learning.
- The language is limited enough so that all parts of the language are derivable. (That is what it means to be a good language for input and output.)

Such an *Interlingua* is required by the learning system to ensure that acquired knowledge can be modified, transferred, or used as the basis for future learning.

Interlingua - a knowledge representation language that, by serving as a standard protocol for the input, output, and configuration bias of a learning process, permits acquired knowledge to be used as the basis for future learning; represents the following types of knowledge: syntactic, logical, procedural, and functional.

The Bootstrapping Process and Bootstrapped Learning Components

It is not expected that all the knowledge acquired by a learning system would fit into a single “theory of everything.” Instead, it is assumed that knowledge which has been bootstrapped is represented as an internally consistent micro-theory expressed in the Interlingua. These self-contained “packets” of knowledge are referred to as *bootstrapped learning components* (BLCs). They first specify the syntactic forms that can be expressed within the micro-theory,

and then indicate specific facts, functions, and procedures expressed in that language, along with heuristics about how to learn them. Thus, a bootstrapped learning process accepts an input composed of BLCs expressed in the Interlingua as well as instructional interaction expressed in the interaction language. Correspondingly, the output of the learning process consists of the refinement and creation of BLCs. This constitutes the basic bootstrapping process.

Bootstrapped Learning Component (BLC) - a self-contained assembly of knowledge expressed in the Interlingua that captures some aspect of a problem domain.

Since it is desirable for learning to occur at a conceptual level appropriate to the domain (e.g., learning how to manage an unmanned aerial vehicle or “UAV” might entail planning knowledge, but probably not the details of real-time flight control), learning is allowed to occur with respect to preexisting software that provides functionality useful in the domain. Thus, a typical BLC has five types of knowledge, which include such software modules:

- A specification of the syntax of knowledge expressible in the BLC’s micro-theory.
- Predefined parameterized software (e.g., hard-coded C process for doing path planning) that carries out actions or computes results in the domain.
- Knowledge that forms a specification of how to use the predefined software and what it does.
- Other initial knowledge relevant to this aspect of the domain (including an appropriate ontology).
- Learned knowledge.

As shown in Figure 3, there are a number of sub-languages of the Interlingua, in which all BLC knowledge is represented:

- hard-coded process, as discussed;
- *syntactic (ontological)*: domain objects and actions, function and predicate types including type restrictions on parameters and return values, etc.;
- *logical*: world knowledge and inference rules;
- *procedural*: knowledge of how to do things in the world; and
- *functional*: knowledge of how to compute complex functions by composing smaller ones.

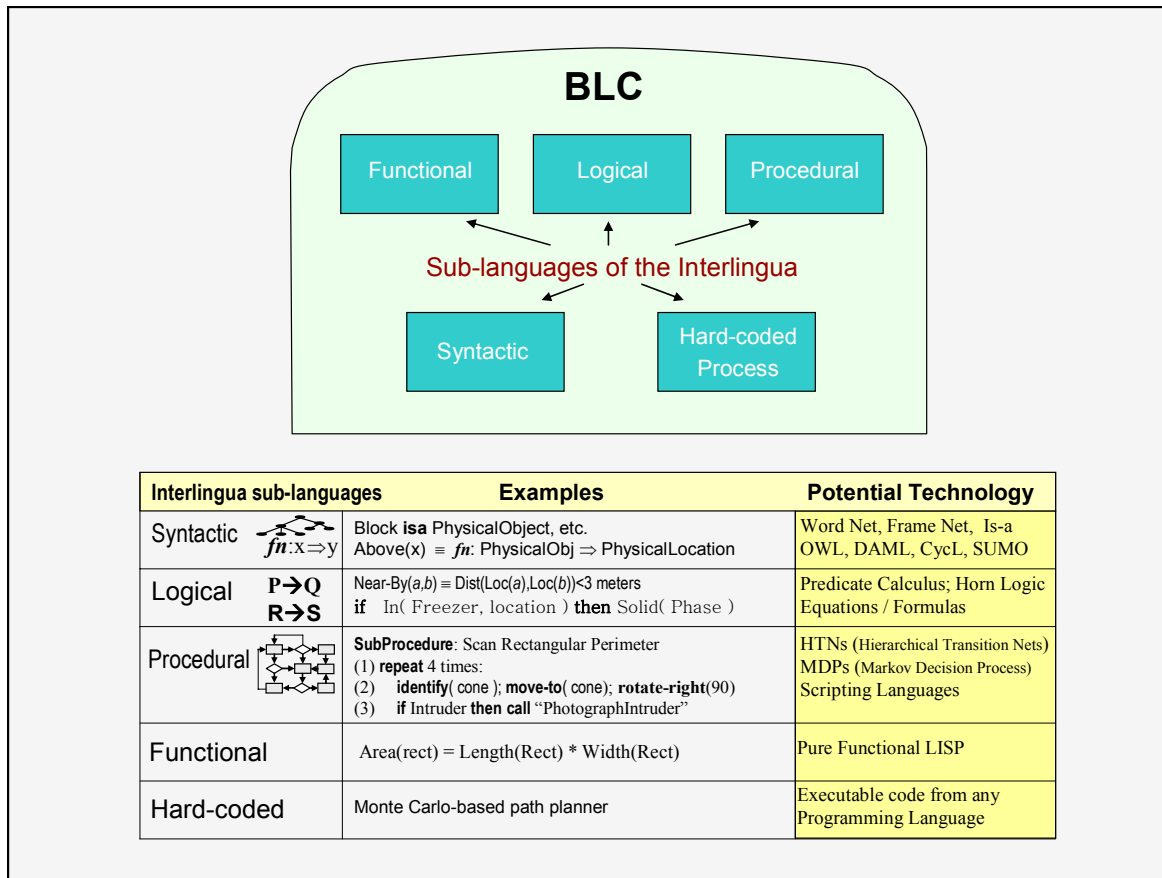


Figure 3: The Structure of a BLC

Here we include several brief examples of BLCs in order to clarify the concept. One BLC might be a hard-coded path planner. Its ontology would provide a set of terms like “goal,” “location,” and “distance,” which can be specialized in different ways depending on their use. Another BLC might provide a “mental” model of the external world (in the curriculum) along with a model of the effect of world actions expressed in that internal mental model. A third BLC could provide a hierarchy of increasingly abstract procedures, which perform actions ultimately in the real world (as contained in the curriculum). In short, any consistently organized body of information can be expressed and, if BL is successful, learned as part of a BLC.

Primed with a deeper understanding of BLCs, it is useful to revisit the functional notion of bootstrapped learning in this richer context. Bootstrapped learning can now be described as a process of building new BLCs from existing BLCs that express a prior, less complete understanding of the task domain for a given curriculum. It would be possible for this process to be used by a learning system that lacks any initial world knowledge to assimilate new curricula. However, in most cases it will be desirable to develop curricula that focus on advanced concepts without having to include instructional content for the low-level, underlying concepts. Thus, it may often be advantageous to bypass prerequisite learning by “injecting” prerequisite knowledge directly into the learning system. This *injected knowledge*, expressed in the Interlingua as one or more BLCs, is provided at the base of a

curriculum ladder and serves as the background knowledge that is required for starting the bootstrapping process. Injected knowledge makes it possible to begin the first lesson of a curriculum at any level in the task domain.

Injected knowledge - knowledge that is provided directly to a learning system, used to prime the entire bootstrapped learning process; includes "common" knowledge and "curriculum-specific" knowledge.

Unlike learning processes, which are domain-independent, injected knowledge can be specific to the curriculum with which it is paired. For example, it might include an appropriate ontology of time or space for reasoning within that curriculum or, perhaps, domain knowledge about the effects of specific actions that exist within a curriculum's world model. This type of injected knowledge is referred to as *curriculum-specific knowledge*.

Curriculum-specific knowledge - knowledge inherent to a specific domain that is a prerequisite for learning in that domain; either taught as a lesson via the interaction language or provided directly (see *injected knowledge*).

In addition to curriculum-specific knowledge, it is expected that there will be classes of prerequisite knowledge that are common to the majority of curricula. These shared BLCs are termed *common knowledge*.

Common knowledge - injected knowledge (BLCs) that are common to most (possibly all) curricula; written in the Interlingua and injected at the beginning of a ladder curriculum.

Most injected knowledge should be of a form that, in principal, could have been bootstrap-learned. However, the problem of Bootstrapped Learning will not be solved in its full generality within the timeframe of this program. Thus, it is acknowledged that some BLCs will have a complexity so great that, in practice, they will not be bootstrappable. It may be intractable, for example, to generate a hard-coded path planning algorithm that employs a simulated annealing Monte Carlo process as the output of a learning process. Accepting this potential limitation, it is nonetheless desirable to minimize the number of injected BLCs that could not have been bootstrap-learned.

To summarize, a BLC is encapsulated knowledge that results from learning and supports the acquisition of higher level knowledge. A BLC may also be introduced directly in the form of injected knowledge in order to provide the learning system with common or curriculum-specific knowledge that is required for successfully completing a lesson. For maximal applicability and reuse, injected BLCs should take a form that could have been generated by a BL process.

Elaboration of the Interaction Specification

As defined, bootstrapped learning depends upon knowledge that is bootstrapped from earlier learning. But it also depends on aspects of the bootstrapping environment that are invariant

and not bootstrapped. The specification of the interaction language is one example of a form of knowledge that underlies BL but is not itself bootstrapped. The ability to identify NI methods as specific instructional behaviors (as opposed to random behaviors) is also assumed knowledge, and is part of the invariant knowledge upon which bootstrapped learning depends. Thus, the term *genetic knowledge* is used to refer to the set of immutable, sometimes implicit, core infrastructure elements, such as those described, that form the basis of bootstrapped learning and permit bootstrapping to occur.

In addition to its invariant nature, another defining characteristic of genetic knowledge pertains to its scope limitations. Genetic knowledge is the minimum set of assumptions to which curriculum developers and bootstrapped learning system developers agree to conform in order to produce a functional ensemble. It is important to note that knowledge common to all curricula (the *common knowledge* listed above) is not considered genetic knowledge, since that knowledge could, in principle, have been bootstrapped.

A goal of the BLP is to identify the genetic knowledge that is sufficient to drive perpetual bootstrapped learning. Genetic knowledge is defined here in terms of its components, which are expected to contribute to that goal:

Genetic knowledge - the invariant, core knowledge that is incorporated into the bootstrapped learning framework; collectively, the interaction language, Interlingua, NI methods, NI method contracts, and semantics of the testing harness that combines the components of the BL framework into a functional, testable platform.

As defined earlier, the interaction specification is the standard by which all interactions between the learning system and its environment are encoded, including interaction with the instructor and with the world. The interaction language is composed of a set of sublanguages, each of which is used to encode abstract instructional communications in one of the defined *interaction modalities*.

Interaction modality - sub-languages within the interaction language. Each is an abstraction of some physical mechanism by which instructional communication may take place (e.g., English speech, written natural language, action in the world, gesture, photograph, diagram, sketch, other perceptual formats, instructional scaffolding, etc.).

Sublanguages for the following four classes of interaction modalities are considered necessary and possibly sufficient to support the goals of the BLP: *world action*, *world perception*, *gesture*, and *linguistic utterance*.

World action - actions that the learning system can perform in the world when practicing, or that it can observe the instructor perform.

World perception - the learning system's perception of the state of the world, or of a hypothetical state of the world (e.g., that an instructor would show as an example).

Gesture - an instructor's pointing gestures to features in other modalities (such as pointing to an action, an object in the world, or a property of an object).

Linguistic utterance - written and/or spoken signals to the student; communicated via a formal, controlled language or something similar to FOPL.

These interaction modalities, in isolation and in combination, are expected to support a spectrum of NI methods. The specific NI methods to be used in the BLP, including the details of how each is specified in terms of interaction modalities, will be agreed upon by all performers. Such an agreement for one NI method is termed an "NI method contract." These agreements will ensure the use of a consistent set of NI methods for interactions between each curriculum and learning system, and within the BLP as a whole.

NI method contract - a formal specification governing student/teacher interaction for a specific NI method; describes the way instructional materials are allowed to be composed using the interaction modalities provided by the interaction language.

Elaboration of the Curriculum

The entire learning context for one topic is represented uniquely by a *laddered curriculum* which contains everything necessary for a student to learn a goal behavior in a given domain. A curriculum, which is specific to a task domain, includes the functionality of a domain-competent instructor, an interactive representation of the world in which the domain is taught, the background and injected knowledge needed for that domain, and a set of lessons.

A curriculum is composed of a set of concept rungs. Each *rung* generates instructional segments expressed in the interaction language. These segments include instructional content in the form of optional live interfaces to the simulated world and live interface to a virtual instructor, generators of initial states, fixed instructional materials, and an exam or other objective assessment of a relevant performance task. It should also be noted that, because a curriculum contains everything necessary for learning a goal task, it implicitly includes the injected knowledge required to initiate the BL process. For testing purposes, each rung will also contain a representation of the goal competency expressed in the Interlingua. This will permit the learning system to skip rungs that it can't learn.

We have identified four principal types of instruction, into which most if not all NI methods fall:

- *Instruction by telling.* The instructor provides general statements that are descriptive in nature, e.g., linguistically. The instructor may combine such utterances with pointing to some aspect of the current state of the world.

- *Instruction by examples.* The instructor may select or present objects or actions in the environment and highlight important features with gestures or annotations. Unlike instruction by telling, this instruction method is completely bound to specific objects or actions known to, and observable by, the student.
- *Instruction using feedback.* When the student provides an answer or solution to a provided problem, the instructor provides feedback that is directly tied to that answer. Feedback is a common method to teach students about special cases or complex metrics such as evaluation functions used to assess the quality of a solution. Practicing to maximize such a function, with or without an instructor present, is a form of feedback instruction.
- *Instructor-guided discovery.* In the course of teaching a lesson, the instructor provides not only material that is relevant to the lesson but indirect instruction about other material that may already be known to the student. For example, in demonstrating how to fly a UAV to a target location, the instruction may include two trucks to be surveyed because they are near each other. Although the point of the lesson is not to teach when two trucks are near, the student should interpret the trucks shown as an example of near and, more specifically, incorporate that aspect of the scenario as including an instance of proximity.

For example, a potential protocol for an NI method in the class *instruction by examples* might be a list of world snapshots (i.e., perception) with optionally-attached linguistic explanation (i.e., relevant sub-expressions) and gesture (e.g., pointing at salient features of the snapshots). An NI method in the class *instruction using feedback* might include a set of practice problems (e.g., world setups) and a specified performance goal expressed either linguistically or during previous learning. This would then allow the student to practice the given task to maximize performance on the goal.

Procedural Overview

A basic understanding of the primary BL framework components, their constituents, and some key functional relationships provides the conceptual basis for a high-level BL walk-through. The following steps refer to Figure 2:

- The learning process begins with injected knowledge in the form of initial BLCs.
- The integration architecture decides which concept rungs to focus on first.
- The curriculum is used to produce the instructional environment (initial world state) and instructional materials (interaction language messages) for a specific instructional experience.
- The integration architecture reasons over the instruction it has received.
- The integration architecture selectively applies one or more learning processes to its input.
- This results in the modification of existing BLCs and the generation of new BLCs.

- For any curriculum rungs that include an exam, the learning system can employ the new BLCs toward achieving the test metrics specified for each rung.

This process repeats, so in BL fashion the BL system learns more and more about the subject, in order to eventually learn the curriculum's goal task, which should result in the BL system attaining *graduate-level performance* in the subject area.

Graduate-level performance - the expected performance level that would be achieved after successful assimilation of a curriculum.

This synopsis of the BL framework serves as a very rough functional overview of framework components and their interrelationships. It does not, however, address programmatic details such as performer roles, teams, and deliverables, which are relegated to Chapters 2 and 3. Furthermore, technical aspects of the framework are discussed in more detail in Appendices B and C, and in evolving external materials, which are described in Appendix A.

Any proposed research should investigate innovative approaches and techniques that lead to or enable the goal of domain-independent bootstrapped learning. Proposals are not limited to the specific program strategies listed above, and alternative visions will be considered. However, proposals should be for research that contributes substantially toward the program's stated goals. Specifically excluded is research that results primarily in minor evolutionary improvement to the existing state of the art or practice, or that focuses on special-purpose systems or narrow applications.

CHAPTER 2. PROGRAMMATIC DETAILS

Phasing

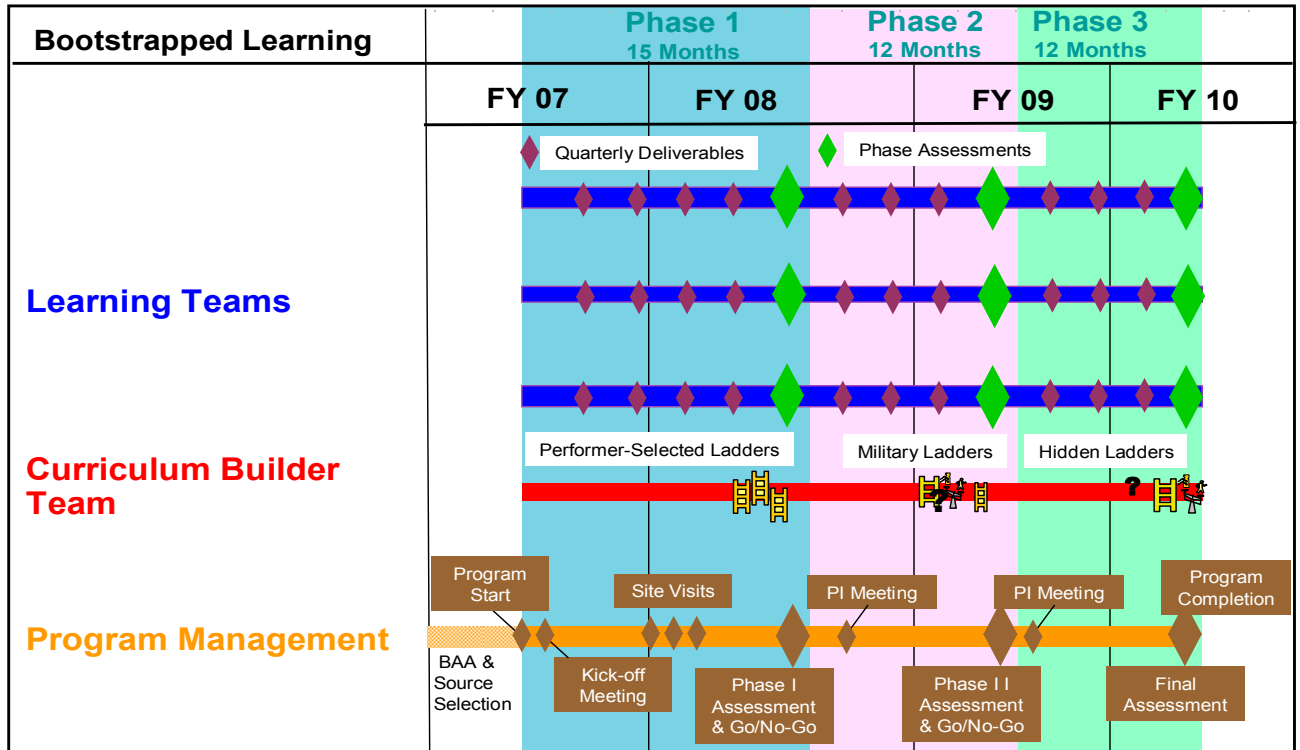


Figure 4: Bootstrapped Learning Schedule

This Program consists of three phases with different time lengths spanning 39 months. The program also includes decision points between each phase, with the decisions based upon the amount of progress made toward the project goals as specified in the proposals and the government-defined go–no go criteria. Proposals should address all three phases. Final decisions regarding the nature (and occurrence) of each phase will be determined by DARPA prior to those phases. The potential phases are listed as: Phase 1 for a 15 month period, Phase 2 (12 months), and Phase 3 (12 months).

Quarterly and monthly deliverables will be extensively employed in the BLP as tight interaction between the curriculum and learning teams will significantly reduce risks to the program goals and improve the efficiency of both teams.

Organization

The BLP solicits research contributions drawn from all applicable fields, including primarily artificial intelligence (particularly machine learning) and cognitive science. It is anticipated that no single approach will provide the desired capability. Instead, an integrative approach that synthesizes new and existing ideas from multiple research areas is more likely to be successful, especially if it is implemented as a set of integrated components within a cohesive theoretical framework.

This program and solicitation calls for two types of proposals: learning team proposals and curriculum team proposals.

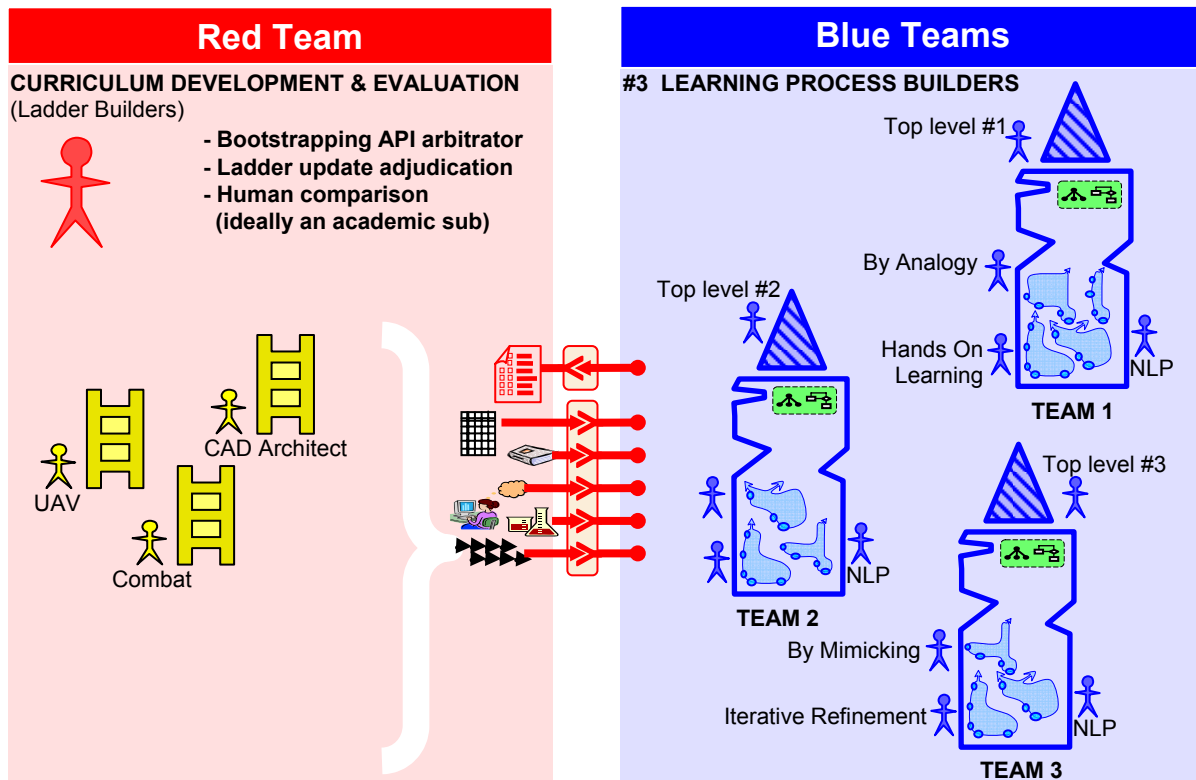


Figure 5: Organization and Structure of Program

Learning System Team

Within the Learning System area, it is anticipated one or more teams will be selected to participate in the program. Each team will construct a self-contained learning system that is capable of learning from a curriculum using the predetermined set of NI methods, and additional methods they may propose. Successful learning teams will need to support a diversity of NI methods within a single system. Thus, it is anticipated that individual researchers will need to collaborate with several others in forming teams for bidding on this aspect of the BLP.

For learning system builders, the BL Program presents an opportunity to perform groundbreaking work in a new research area. Because the curricula will be built by a separate performer, Learning System developers will be able to concentrate predominantly on research while still being able to perform rigorous experiments across multiple diverse test domains. The existence of a formal, high-level, standardized interaction specification for this new rich data source (the curricula) ensures the learning teams can focus directly on the learning research. The support of each NI method will constitute a novel research problem. In most cases, the instructor interaction will represent a source of data that does not exist today. Thus, building a BL system will provide fertile ground for many sub-components that are the first of their kind. Interesting additional research possibilities related to the interactions that occur among NI methods are also anticipated.

Curriculum Team

For the Curriculum area it is anticipated that one team will be selected for the program. The Curriculum team will provide at least three curricula per phase for multiple domains. Each curriculum will include domain-specific “injected” knowledge comparable to the prerequisite knowledge that a human student would possess (see Appendix C). This assumed knowledge would be explicitly imparted to the electronic student prior to learning. In addition, each curriculum will include one or more domain simulators which provide “live” interaction with the world, an “electronic instructor” specialized to the relevant domain, and structured instructional materials.

Domain Specialists

In order to provide a context consistent with program goals, DARPA has indicated general areas of interest (see the section on “Curriculum Ladders” in Chapter 3). Within those areas, domains can be proposed by potential performers, permitting them to exploit their existing resources. Due to the expected variety of domains, a successful curriculum team will likely be composed of multiple organizations, each providing a portion of the expertise required for the diverse curricula.

Genetic Knowledge

Another important additional role of the Curriculum development team will be the arbitration of issues related to defining the Learning System’s *genetic knowledge*. In particular, the Curriculum team will work with the learning team(s) under the direction of DARPA to arrive at a consensus.

The Learning System and Curriculum are elaborated in Appendix C, and related team roles described in Chapter 3.

Awards

The dollar amount of awards will be determined by the quality of proposals and funds available.

Learning System Development – DARPA anticipates awards to one or more integrated team(s), each of which would take alternative approaches to achieving the desired technical capabilities necessary for realizing Bootstrapped Learning. Teams may have any organizational structure capable of performing the work.

Annual funding for each team is anticipated to be commensurate with the goals of the program. Smaller groups are encouraged to collaborate to form complete BL learning teams.

Curriculum Development – DARPA anticipates providing an award to a single curriculum development team.

CHAPTER 3. PERFORMER REQUIREMENTS

Introduction

Figure 2, introduced in Chapter 1, shows more detail of the general approach of the program by focusing in on a single curriculum or ladder. It is important to emphasize that, in general, the learning developers will not know what these ladder domains are while developing the learning system. In addition, in at least one domain, the curriculum developer will compare the performance of the learning systems to human learning on the same curriculum.

Requirements of the Learning Team

This subsection discusses the requirements for the learning team role. The main task of this performer is to build a general-purpose “electronic student” that bootstraps complex behaviors from natural instruction (NI) lessons.

The BLP anticipates funding one or more learning team(s). Each team will construct a self-contained learning system that is capable of learning from a curriculum built according to a given set of NI method contracts. It is anticipated that each team will include a variety of performers with different skills who will collaborate to create the learning processes, student infrastructure, and top-level control strategy. Learning team proposers will enumerate specific NI methods and, during the program, develop new learning algorithms for learning from them. Each of these performers must have a very strong track record in a field related to the specific NI method it is targeting and must propose innovative algorithmic ideas for how to efficiently perform learning given the inputs for the NI method. The bidder may also provide innovative instructional scaffolding that is both realistic and improves performance considerably.

Integration Architecture

The integration architecture, with its ancillary reasoning capability, is key to a successful learning system. This architecture must:

- accept and reason about natural instruction (NI) steps from the current curriculum, expressed in the interaction language
- provide outputs to the instructor or world in the interaction language
- reason over the Interlingua, including encoded knowledge in the BLCs, about what has been learned and is to be learned next, and which NI methods should be applied.
- translate the instructional material into a form that is consistent with the NI method contract. For example, limited inference might be required to map literal percepts onto the knowledge required by the NI method contract. This inference would be executed and controlled by the integration architecture.
- determine when each learning process should be invoked, and with what inputs

Learning Processes

A central goal of the BLP is the development of new learning processes. Each one will be specialized to a particular interaction with the instructor (i.e., to a particular NI method contract) instead of being specialized to a particular problem domain, which is typical in ML today. Furthermore, existing domain-independent ML does not exploit learning methods that benefit from the richness of instructor interactions, such as those supported by the interaction language.

Each learning team will propose a set of NI methods for which its proposed learning processes are best suited. Based on these proposals, DARPA will select a common set of NI methods that the BLP will support. Under the stewardship of the curriculum team, a common set of NI method contracts will be adopted in the early months of the program, and will be adapted throughout the program. Each team will be responsible for supporting the range of instruction methods implied by these contracts.

Desired Characteristics of a Learning System Developer

A successful learning team will have:

- capability to develop new, or modify existing learning algorithms to operate in the BL paradigm with inputs according to specific NI method contracts
- capability to create a novel controller (integration architecture) with the capabilities listed above
- some team members with very strong backgrounds in formal and practical AI algorithm development

DARPA would like the prime contractor of any learning team bidder to have staff with experience in program management and in developing and integrating complex research software prototypes into a system. This experience should be used to create a program plan and schedule, and to implement mechanisms and procedures to drive the integration of the various learning processes. It is also desirable that the staff of any prime contractor have experience with machine learning research as well as other AI techniques. Developers of learning processes should be recognized leaders in the fields of research they bring to the BL program. In particular, they should have significant practical experience developing and experimenting with such algorithms.

Key Technical Questions to Be Addressed by a Learning Team Bidder

A bidder for a role as learning team should explain what is novel about its approach, why it will solve the difficult technical problems stated in or implied in the preceding discussions, and why it will be successful in contributing to the two major scientific objectives of the BLP:

- *Claim #1: domain independence.* A *fixed* BL-based system is capable of learning a wide range of performance tasks, in the form of curriculum ladders, in a wide range of problem domains, based on abstracted NI methods, with *no* reprogramming or reconfiguration of the learning system between learning sessions.

- Claim #2: *good performance*. The bootstrapped learning performance of such a system compares reasonably well with human learning performance, as defined by the evaluation criteria laid out in the section below, titled “Formal Evaluations of the Learning Systems.” Furthermore, this comparison holds when each system (BL and human) is provided with the “same” background knowledge followed by the “same” instructional content using the same NI methods (see discussion of human comparison for details on this).

For the integration architecture proposed, the following should be addressed:

- innovative claims about the architecture
- how the control regime will allow and make learners work together
- how the control regime will use introspective and other ancillary methods in conjunction with the learning methods to address the instructional material
- issues of algorithmic robustness
- robustness to noise
- robustness to missing information (no pre-requisites)
- algorithmic efficiency

For each learning process or group of processes being proposed, the following should be addressed:

- the type of natural instruction being proposed
- if the NI method is new, why this type of NI is practical and general
- how this NI method integrates with other methods
- the technical approach to learning (algorithmic approaches) being considered
- the innovative claims for this learning process
- how this approach improves on existing learning technology to work in bootstrapping
- how this approach tractably integrates the multiple constraints provided to each NI method
- how bootstrapping can be repeated indefinitely
- how shifts in representation and learning bias can occur when appropriate

In general, it is the prime contractor’s end-to-end capability that matters most. Thus, the prime contractor should articulate clearly how all performers’ will be integrated tightly to produce a working, robust BL system.

Requirements of the Curriculum Team

This subsection discusses the requirements for the curriculum team role. The curriculum team's primary responsibilities in the BLP are (a) to coordinate, with the learning teams, the creation and maintenance of the genetic knowledge necessary for a learning system to learn in a range of domains; (b) to generate curricula complete enough to teach a learning system about the selected performance tasks; and (c) to conduct the formal progress evaluations at the end of each program phase. There will be one curriculum team.

It is anticipated that the curriculum team will consist of performers that provide a variety of technical skills and domain expertise. These include coordination of a multi-team software, language, and interface specification effort, software integration, interface specification, knowledge engineering, educational curriculum development, providing human expertise in the selected domains, selecting and integrating simulations in the selected domains, conducting formal system evaluations, and conducting evaluations of human versus machine learning performance.

Genetic Knowledge

The curriculum team is the steward of the genetic knowledge underlying the BLP. The team must have personnel specifically qualified and tasked to collaborate with the learning teams through workshops and other forums they organize on the refinement and maintenance of these languages and frameworks. In particular, the curriculum team should identify a key staff member who will oversee the negotiations over these languages. This individual should be a leader with significant experience developing technical languages.

The genetic knowledge of the BLP includes the interaction language, NI methods, and Interlingua language specifications, the format for BLCs, and the testing framework. Additionally many pieces of common knowledge will also need to be developed with input from the learning teams, including all prior knowledge needed to bootstrap learn the curricula. A draft specification of a possible interaction language is available at the "Interaction Language" link on the BLP website listed in the External Materials section in Appendix A. Also available on the website is a current specification of a possible Interlingua as well as a discussion of the initial taxonomy of NI methods (follow the "Natural Instruction" link).

The curriculum team and the learning team must all use the same genetic knowledge. In order to finalize these elements after all performers have been selected, the curriculum team will lead and coordinate the evolution of the specification of each element of genetic knowledge. This task may include drafting written specifications and selecting existing standards. It will also including organizing workshops at the start of the BLP to discuss and develop an agreed upon specification of all of the types of genetic knowledge that will be employed. DARPA will have final approval of all decisions. Once a decision has been made, the genetic knowledge specifications will be changed only as a result of demonstrated necessity, since changes would affect both the learning developers and curriculum developer.

In this context, the NI methods are of particular note. The curriculum team will create the lesson rungs of the curricula, the choice however of which NI method to use for each lesson, will depend upon the needs of the final domains selected by DARPA, and the testing needs for NI methods.

As described in the section on curriculum team deliverables in Chapter 7, the curriculum team will be responsible for providing at least one example lesson or “vignette” for each new NI method contract which serves as a stable working example of the instructor-based execution of that NI method. This vignette must be produced for the learning teams soon after the NI method contract is agreed upon. The curriculum team may arrange for that vignette to be reused as part of the curricula that are delivered later.

NI Method Contract Assessments

It is also the responsibility of the curriculum team to ensure the plausibility of the NI method contracts. This will be accomplished through consultation with appropriate researchers (e.g. educational psychologist, or other experts) to perform relevant assessments of the contracts. These assessments will be performed in an intermittent, targeted manner as dictated by the phasing structure and deliverables schedule of the program. The researcher will evaluate the plausibility of each NI method contract in terms of its correspondence to the actual capabilities and behaviors of human students and instructors. The researcher will also provide guidance on how to evolve the program to maximize the yield of instructable computing technology. Toward these ends, the curriculum team will deliver written assessments of the validity and utility of NI method contracts.

Bootstrapped Learning Support Infrastructure

The BL support infrastructure includes the physical embodiment of the genetic knowledge (e.g., parsers, interpreters, etc.) as well as a testing harness that provides support for empirical research (by varying parameters, cross-validated testing, etc.), for use in formal evaluation and in research by the learning teams. The curriculum team will be the steward of this infrastructure. In particular, it will accept and maintain current versions of the learning systems, curriculums, and framework in order to make available, in an ongoing manner, a package that represents the current assets of the BLP. The associated deliverables are detailed in Chapter 7.

The curriculum team will create the genetic knowledge that each learning system will possess. It is responsible for procuring, modifying, and/or maintaining, as necessary, relevant existing languages, tools, software packages, APIs, ontologies, and knowledge bases. The goal of having such a “shrink-wrapped” version is to permit individual researchers on the learning teams to easily download a complete testing platform that would include multiple ladders and versions of multiple teams’ learning systems. This would allow him or her to develop and test specific algorithms within that context.

Curriculum Ladders

Types of Curriculum Ladders

Most of the curricula generated in the BLP are called *diversity ladders*. These are used to ensure the generality of the learning systems. Development and refinement of the learning systems is driven by these ladders. However, phases II and III in the program will call for three specialized types of curriculum ladders: hidden, military transfer, and human comparison ladders:

- *Hidden ladders* – these ladders only become known to the learning teams after they have frozen their systems for testing. Proposers agree that any candidate curriculum team that has identified a potential hidden domain in its proposal will conceal the domain from any potential learning teams. Furthermore, proposers (in their BAA response) should explain what, if any, knowledge about the ladder may be available to learning team proposers.
- *Military transfer ladders* – these ladders will provide scenarios that have military relevance; in the limit, such a ladder could be used to instruct a learning system that would then be field-ready for military use. In other words, a learning system taught by a military transfer ladder, equipped with sensors, actuators, and fixed algorithms to perform perceptual abstraction, would be an end-to-end trainable solution to an important military problem. Teams should propose as many military ladders as they can without compromising other aspects of the program.
- *Human comparison ladder* – one ladder will be a human comparison ladder, to be used for assessing the performance of the learning system with actual human performance on the same curriculum. Toward this end, the ladder will be “decompiled” back into human viewable form (or encoded in two forms).

Ideally, a single ladder would embody the characteristics of all three specialized ladder types. However, there may be technical reasons for pursuing other arrangements. Nonetheless, the curriculum team must provide all types of ladders whether or not they are combined, and they should indicate which ladders they believe (1) have potential military transition, (2) could be used in a human comparison, and (3) could serve as the “hidden” test ladder at the end of phase II and III.

Curriculum Ladder Domains

The curriculum developer will create a variety of automated NI methods and then use them in creating a number of complete curricula. A *complete curriculum ladder* consists of:

- The assumed domain-specific injected knowledge, expressed in the Interlingua
- A simulator for the domain, to allow the student to interact with the outside world and for testing

- Generators for initial states for each of the instructional episodes, as well as an electronic teacher that provides interaction as needed for that episode. *The goal is to provide effective teacher interaction for each lesson with minimal cost, even if this means providing a very domain-specific (narrow) solution for providing interaction.*
- For the one “human comparison” ladder, an interface to the simulator and all training materials usable by either an electronic learning system or human student.

During the three possible phases of the program it is expected that research will be conducted over multiple curricula in multiple domain areas. The curriculum developer will be required to develop at least one complete curriculum for each of the selected domains. These complete curricula will then evolve over the lifespan of the program. The goal for the curricula collectively is to exercise the diversity of the NI-method contracts, and provide compelling evidence that BL can be used as a means of delivering instructable computing systems in important domains.

Over the life of the BLP, new curricula will sometimes involve entirely new domains, simulators, etc. (as is the case with the hidden curricula.) In other cases, they will involve reuse of generators, simulators, and many of the other components appropriate for a given domain.

Based upon their expertise and existing capabilities, curriculum team bidders should propose at least three curriculum domain options from the following broad subject matter areas listed below. Each curriculum domain option should specify the simulator to use (a domain), and one or more curricula scenarios detailing what is to be learned. Bidders should express their proposal as a base cost (for the core infrastructure, human testing, etc.), plus multiple curriculum domain options. As a very rough guide DARPA offers three as a possible number of options to be exercised. But this is only a guide; many factors will contribute to the ultimate decision which could be quite different. (See Chapter 7, Section IV-Cost Volume) In order to reduce time, cost, and risk, a bidder should take advantage of assets developed outside the BLP as much as feasible. Diversity and number of proposed curricula domains is a consideration when evaluating the curricula team. Although a curriculum development bidder should propose a team that can deliver complete curricula for the domains it proposes, it should not commit funds to its domain-specific performers until the domains have been selected by DARPA after the start of the BLP. Over the course of the BLP many constraints will be considered in directing the curriculum work. The broad areas of interest are:

- **Automation** (autonomous robotic systems, autonomous software agents) – Many existing and emerging systems could be “programmed” using BL. Each curriculum team must include at least one curriculum in this area, which includes autonomous vehicles, surveillance systems, situation assessment/anomaly detection and many other configurable systems.
- **Strategy Games** – Games where one could instruct humans and machines to employ the multi-level reactive strategies needed to win (examples are Sim-City and many war games)
- **Authoring Simulation Agents or Authoring Tutoring Systems** – Use of BL technology to author simulated agents or authoring tutoring systems. For example, a

BL system could learn to perform some designated role (like Platoon Leader) within OneSAF, the Army's standard semi-automated forces simulation system for training and analysis; or a BL system could learn to be a *teacher* within the OneSAF world, that is, it could learn what types of feedback it should provide when it observes particular failures in some student it is observing.

- **Planning/Scheduling** – Use of BL to produce planning and scheduling systems (e.g., learn to produce resource allocation plans, or battle plans given a set of constraints).
- **Diagnosis** – Use of BL to produce diagnostic systems (e.g. learning strategies to investigate, diagnose, and correct failures at a nuclear power plant).
- **Design** – Use of BL to learn new expert systems for specific design tasks (e.g., building layout or electronic filter design).

Teams should provide costing options for at least three curriculum domain options. At least one of these options should be from the automation area. Each option should be for a specific simulated domain and at least one specific training curriculum scenario (e.g., teaching pawn advancement strategies in chess), per phase. At the start of the program, and in negotiation with the learning teams and DARPA, a final choice on phase one curriculum domain options will be made. In subsequent phases, additional curricula would be delivered for each of the domains selected in Phase One. In addition to the options above, the curriculum teams should also provide options for at least two hidden domains as well. (See Chapter 7, Section IV – detailed cost breakdown.) Unlike the curriculum domain options these options should only provide funding for the domain simulator, and a single test curriculum (unlike the curriculum domain options, the hidden domain options could be executed within a single phase.)

The ultimate goal for the curriculum team is to provide the most curricula at the lowest cost in a manner suitable for cross-validated testing of the BL learning systems. To facilitate this, curriculum teams should provide a base capability for generating curricula that can be redirected by DARPA over the course of the program as needed to study the most important aspects of bootstrapped learning. Thus, it is preferable that multiple curriculum domain options have the exact same costing if possible, this allows option selection without affecting total cost. Additionally, to minimize costs, teams should leverage the reuse of existing software assets, automation of initial state generation, automation of instructor feedback, and scenario generators. Curriculum options and cost-mitigation strategies should be addressed in the proposal as indicated in Chapter 7, in the section titled “Curriculum Team Deliverables.”

Any NI method contracts can specify student-initiation, teacher-initiation, or a mixed-initiation approach. NI methods that are heavily biased toward teacher-initiated instruction will place a greater burden on the development of electronic instructors. Thus, such methods will require a proposal (*from the proposing learning teams*) on how such interactions can be provided using an automated scheme in a cost-effective way.

The Electronic Instructor

It is recognized that a functional symmetry exists between students and teachers. They are both agents of bi-directional communication, possess knowledge and meta-knowledge, and

are goal-directed. Even their absolute goal is the same (i.e., the transfer of knowledge from the instructor to the student). It is only their respective roles in achieving this goal that distinguishes them.

In the context of BL, the details of the roles distinguishing teacher and student will be dictated by each NI method. A natural consequence of having different roles is that in order to perform effectively, teachers and students will necessarily benefit from different core competencies. Nonetheless, it is commonly accepted that learning is most efficient in a mixed-initiative instructional approach, which involves the student directing the teacher, and the teacher acquiring an understanding of the student's deficits from the student. Thus, one could argue that in an optimal learning context, there would be significant overlap among the core competencies of students and teachers.

If one envisions electronic students and electronic teachers as derivations of a more general class of "electronic agents," it is conceivable that shared competencies underlying both learning and teaching could be represented as genetic knowledge in the form of BLCs that are shared across electronic teachers and students. It is further imaginable that a *dissemination architecture* analogous to the student's integration architecture could transform BLCs into lessons articulated in the interaction language. In fact, if one subscribes fully to the symmetry model, the teacher and the student would each possess both an integration architecture and a dissemination architecture.

Though the development of an electronic teacher is not an articulated goal of the program, it is in some sense an implicit one in that, on pragmatic grounds, there is the need to automate the instruction of the electronic student. Thus, there is potentially significant development time to be saved by adopting such an "electronic instructor paradigm" in which the curriculum team develops a multi-use electronic agent that can be used to efficiently produce instructional interaction from existing parts of the curriculum. For example, consider that each curriculum rung is required to have an "answer" encoded in the Interlingua upon which the student can rely if its learning fails on that particular rung. One could imagine building an automated instructor which produced detailed feedback for the student, based upon the student's behavior and the curriculum's encoding of the answer. When the student makes predictions that do not vary in response to a feature that, according to the answer expression is important, an automated instructor could gesture at that feature as being relevant. As a second example, consider that the curriculum team will be responsible for significant pieces of domain knowledge in the form of injected BLCs for each domain. These components could potentially be used by an automated instructor process in generating explanations or other feedback to the student, again in an automated fashion.

Taking this approach is not a requirement of the program. In fact, there are potential risks associated with this approach, such as the incremental time necessary to develop suitable general purpose methods. Nonetheless, innovative ways to *reduce total cost for curriculum development* through automation or by other means, should be an aspect of curriculum team proposals.

Injected Knowledge

The curriculum developer will create, through knowledge engineering and software integration, the injected knowledge for each selected domain, in the form of BLCs. Existing software components that provide functionality useful in the domain will be created/selected and integrated. Knowledge engineering, using the Interlingua, will result in a domain-specific ontology, the initial domain knowledge, and specifications of the software components for each curriculum.

Spatial Reasoning BLC

The need for spatial reasoning is present in all of the domains considered to date. Thus, DARPA is open to some research within the curriculum team toward developing a BLC that provides spatial reasoning capabilities. This BLC would serve as a starting point for bootstrapping in domains that require spatial reasoning. The spatial reasoning BLC could include: a spatial ontology, a quantitative 3D reasoner, a qualitative (e.g., linguistic) spatial reasoner, a topological reasoner, and mappers that convert among the respective representations of these three reasoners (i.e., 3D, qualitative, and topological).

The researchers who develop the spatial BLC would need to be involved in the development of curricula consistent with this background knowledge. Furthermore, this BLC would have to be implemented in tight collaboration with the learning teams in order for it to be of use during learning. Since the BLP is *not* focused on spatial reasoning, the total cost of this aspect of the curricula work should be minimized. Reuse of existing ontologies, reasoning systems, and other applicable constructs is encouraged when possible.

The other injected knowledge BLCs will be selected later, since they depend on the domains that are chosen for the program. However, a curriculum team bidder should propose candidate BLCs as required for the task curricula they propose.

Formal Evaluations

The curriculum team will accept the developed learning systems and experiment with them, conducting various tests and evaluations. The formal evaluation protocols are defined in the subsection below titled “Formal Evaluations of the Learning Systems.” One domain will be selected for a head-to-head comparison of the electronic student and a set of human students who will learn via the same ladder expressed through the same interaction language. In practice, the same interaction language specification used for the electronic student will be “decompiled” manually or automatically back into a human viewable format in order to conduct the human testing. The curriculum team bidder should propose how best to do this. Finally, the curriculum team will conduct a statistical analysis and assessment of the results of the formal evaluations.

Mitigation Strategy for Potential Conflict of Interest

A primary concern with respect to teaming is to avoid a conflict of interest (as explained in FAR 9.5) that could result in contamination of the go/no-go evaluations. Thus, it is not

permissible to be a prime for both the curriculum team and a learning team. Additionally it is the responsibility of the curriculum team to guarantee the integrity of the testing provided by its curriculum ladders and by its evaluation process. In particular, it must ensure that sub-performers will not be in a position to control important aspects of the tests that they, or those with whom they are affiliated, will subsequently be receiving in a learning team role. Any inputs that are provided by sources with possible conflicts must be cleansed in some way that ensures the integrity of the evaluations. The curriculum team must articulate in its proposal how it will ensure that the integrity of its evaluations is maintained.

Desired Characteristics of a Curriculum Developer

A successful curriculum developer will have significant expertise in the following areas:

- managing complex software integration projects
- knowledge engineering
- software engineering
- tutoring system / educational system development
- evaluation of learning

More extensively, a successful curriculum developer will have knowledge of:

- educational curriculum development
- complex languages, tools, and APIs
- knowledge representation, reasoning, and ontology languages, tools, and techniques
- knowledge engineering
- interface specification
- software integration
- selecting, extending, and integrating simulations
- coordination of a multi-team software, language, and interface specification efforts
- formal evaluation of the performance of AI systems
- human subject testing

Furthermore, a successful curriculum developer will have the specific capabilities to:

- select and integrate a general upper-level ontology
- create a variety of automated curriculum generators
- provide domain expertise in a variety of areas
- create one or more curriculum ladders that utilize the NI methods to teach a single subject
- create curriculum ladders for all domains chosen by DARPA

- use knowledge engineering to create the assumed prior knowledge for each domain, expressed in the Interlingua
- provide and integrate an appropriately realistic simulator for each domain
- lead and coordinate the definition, implementation, and maintenance of a complex interaction language that can support interactions with either a learning system or a human student
- provide two “equivalent” interfaces to the simulator, one for use by a learning system and the other by human students (note: this is only needed for the one “human comparison” ladder)
- provide training to learning teams and to human students
- lead and interact with the learning team contractors; evaluate their systems via experimentation and testing
- lead workshops and other forums to develop/refine the various languages of the BLP (the genetic knowledge used in the program)
- conduct various forms of testing, experimentation, and evaluation of human subjects’ ability to learn via the interaction language
- perform a statistical analysis and assessment of the results of the formal evaluations

Key Technical Questions to Be Addressed by a Curriculum Developer Bidder

A bidder for the role of curriculum team should explain what is novel about its approach, why it will solve the difficult technical problems raised or implied by the previous discussions, and why it will be successful in contributing to the two major scientific objectives of the BLP:

- Claim #1: *domain independence*. A BL-based system is capable of learning a wide range of performance tasks, in the form of curriculum ladders, in a wide range of problem domains, based on abstracted NI, with *no* reprogramming of the learning system between learning sessions.
- Claim #2: *good performance*. The bootstrapped learning performance of such a system compares reasonably well with human learning performance, as defined by the evaluation criteria laid out in the section below, titled “Formal Evaluation of the Learning Systems”. Furthermore, this comparison holds when each system (BL and human) is provided with the “same” background knowledge followed by the “same” instructional content using the same NI methods.

Formal Evaluations of the Learning Systems

How the Claims Will Be Tested

The evaluation method for the learning systems is derived from the principal claims that are being asserted:

- Claim #1: *domain independence*. A BL-based system is capable of learning a wide range of performance tasks, in the form of curriculum ladders, in a wide range of

problem domains, based on abstracted NI, with *no* reprogramming of the learning system between learning sessions.

- Claim #2: *good performance*. The BL accomplished by such a system compares reasonably well with learning by a human, when each is given the “same” background knowledge followed by the “same” instructional content using the same NI methods.

Claim #1 Assessment: Domain independence (i.e., generality) will be tested by executing learning trials over multiple curriculum ladders in various domains. During testing, the learning system will not be modified or reconfigured. At least one of the ladders will be a hidden ladder.

Claim #2 Assessment: Performance with respect to humans will be tested by performing head-to-head comparisons between a learning system and humans. A single curricula ladder will be provided (in appropriate human readable and machine readable formats to both human subjects and the BL learning systems). Details of how this comparison will be performed is specified in the BL Metric section.

The Assessment Protocol

The learning systems developed in the BLP will be evaluated quantitatively at the end of each phase according to a fixed protocol, which applies to a single learning system that is receiving instruction from a single curriculum. Each concept rung in a ladder curriculum includes an injectable representation of the “answer knowledge”—the knowledge that is expected to result from complete assimilation of the target concept. The assessment protocol specifies specific performance thresholds as a function of the fraction of these “answers” required by the BL learning system needed to achieve a given level of performance.

Specifically, a parameter, called “percent solo, or attempted to solve unassisted” and abbreviated “%s”, stipulates the minimum percentage of concept rungs in a curriculum ladder for which learning must be attempted without the option to “look up the answer” by receiving the concept as injected knowledge. Inversely, $(1 - \%s)$ dictates the maximum percentage of concept rungs for which the learning system may “divine” the target concept via injection. For example, consider a curriculum with ten concept rungs. If an assessment is conducted with %s set at 60, the learning system would only be able to “lookup the answer” on a maximum of four (40%) concept rungs. Similarly, if %s is set to 100, then no answers may be “looked up” during learning. Once the %s parameter has been set, the learning system attempts to learn the curriculum. Performance is measured using an objective function (e.g., in a situation awareness domain, the number of enemy combatants located within some fixed amount of time) associated with the top rung of the curriculum ladder,

Recall that for the human comparison ladders, learning system performance will be assessed relative to the performance of a group of human subjects learning from exactly the same curriculum. Although efforts will be made to select domains and human subjects such that both machine and humans start out with the same genetic knowledge, there will undoubtedly be additional useful knowledge that the humans possess that the machine does not. To compensate for this issue a comparative measurement of improvement will be used. The

improvement-based metric will be defined as the difference in scores on a standardized test (i.e., the objective function) in the domain taken after the entire curriculum ladder has been learned, and before any exposure to the curriculum ladder. This difference is referred to as “ P_{Δ} ”. Thus, the relative improvement of the learning system will be computed as the ratio of learning system improvement to the average human improvement and referred to as “percent improvement” and abbreviated “%p”.

$P = \text{objective function output}$

$$P_{\Delta} = P_{\text{post-learning}} - P_{\text{pre-learning}}$$

$$\%p = \frac{P_{\Delta}(\text{learning system})}{E[\%P_{\Delta}(\text{human})]}$$

Figure 6: The BL Metrics

For the assessment of learning on diversity ladders, performance will be computed as a ratio, just as in the case of the human comparison ladder, but the curriculum ladder author will specify a graduate level that is used as a reference to compute the percentage (%p).

In order to derive a single composite index that represents learning achievement across multiple curriculum ladders and to focus the research on the areas of poorest performance, the *minimum* score obtained for all ladders will be used.

Go/No-Go Tests

The Phase 1 go/no-go test is designed to assess development progress, not performance, and hence does not rely upon the assessment metric. Instead, the learning team will test its learning system on a curriculum ladder that it has defined and built with DARPA’s approval. The curriculum team will verify that, during the test, the learning system employed no fewer than three different input modalities of the interaction language, two different learning processes, and three rungs of the ladder.

In phases 2 and 3 the assessment protocol and metrics will be used, and the curriculum team will propose the test problems (with DARPA’s approval). The minimum parameter values (%s) and scores (%p) for a “go” decision are provided in Table 1.

The *graduate-level performance* of a curriculum is defined as the expected performance level that would be achieved by a student who fully assimilates a ladder for that curriculum. The curriculum team and its domain experts will decide upon a benchmark value for each ladder curriculum that reflects the level of performance that would result from complete assimilation of the curriculum. This benchmark will be utilized in evaluating all of the ladders that are taught only to the learning systems. However, for the ladder that is taught to both learning systems and humans, the “graduate performance” will be set to the average human performance improvement, based upon experimental trials.

Phase	Claim #1: Domain Independence	Claim #2: Good Performance
2	75% <i>s</i> & 90% <i>p</i> of graduate-level performance across the ladders produced by the curriculum team in phases one and two (except the hidden ladder)	50% <i>s</i> & 70% <i>p</i> on hidden, human-comparison ladder
3	90% <i>s</i> & 90% <i>p</i> of graduate-level performance across all ladders produced in the program (except the hidden ladder)	80% <i>s</i> & 80% <i>p</i> on different hidden, human-comparison ladder

Table 1: Go/No-Go Thresholds for Phases 2 and 3

Additional Tests

In addition to the formal go/no-go tests, during phases 2 and 3 one metric will be evaluated that will not be used to determine go or no-go of the program. This metric is an attempt to estimate how much BL reduces field-training costs. The field-training cost of a particular curriculum ladder learned by a particular learning system is defined to be the *estimated* cost in person-hours to generate the training data required by the learning system to reach graduate-level performance on that ladder. The reduction in field-training costs is then defined as the ratio of the field-training cost of the current system to learn a ladder to the cost of a traditional software development cycle aimed delivering the same capabilities. A host of other metrics that measure the performance contribution of each learning algorithm are also of interest.

Details of Human Testing

Some details of how the testing of a curriculum with humans will be conducted:

- All student instruction will be provided directly from the computer (using the same curriculum ladder given to the learning system, but via a different API).
- The top rung of each ladder will have an “examination” consisting of a problem generator and scoring function with which to test overall proficiency.
- In order to establish the level of improvement, a student will be tested prior to any instruction in the new domain and then tested again after learning from the entire ladder.
- All go/no-go thresholds must be achieved with high confidence ($P > 95\%$).
- Since each student’s performance is independent, a single-tailed t-test will be used to assess confidence. 20 to 40 test subjects should suffice to achieve this confidence level.

CHAPTER 4. GENERAL INFORMATION

This notice, in conjunction with the BAA 07-04 FBO Announcement and all references, constitutes the total BAA. No additional information is available, nor will a formal Request for Proposal (RFP) or other solicitation regarding this announcement be issued. Requests for same will be disregarded.

Web site, ongoing Q&A and other external information

The solicitation web page at www.darpa.mil/ipto/solicitations/solicitations.htm will have a great deal of information including the previously released RFI and the proceedings of the RFI workshop. A Frequently Asked Questions (FAQ) list may be provided there, as well.

Offeror eligibility

All responsible sources capable of satisfying the Government's needs may submit a proposal that shall be considered by DARPA. Historically Black Colleges and Universities (HBCUs), Small Disadvantaged Businesses and Minority Institutions (MIs) are encouraged to submit proposals and join others in submitting proposals. However, no portion of this announcement will be set aside for Small Disadvantaged Business, HBCU and MI participation due to the impracticality of reserving discrete or severable areas of this research for exclusive competition among these entities. Independent proposals from Government/National laboratories may be subject to applicable direct competition limitations, though certain Federally Funded Research and Development Centers are excepted per P.L. 103-337§ 217 and P.L 105-261 § 3136.

Foreign participants and/or individuals may participate to the extent that such participants comply with any necessary Non-Disclosure Agreements, Security Regulations, Export Laws, and other governing statutes applicable under the circumstances.

Administrative Notes

Restrictive notices notwithstanding, proposals may be handled for administrative purposes by support contractors. These support contractors are prohibited from competition in DARPA technical research and are bound by appropriate non-disclosure requirements.

Subject to the restrictions set forth in FAR Subpart 37.203(d), input on technical aspects of the proposals may be solicited by DARPA from non-Government consultants /experts who are strictly bound by the appropriate non-disclosure requirements.

It is the policy of DARPA to treat all proposals as competitive information and to disclose their contents only for the purpose of evaluation. No proposals will be returned. Upon completion of the source selection process, the original of each proposal received will be retained at DARPA and all other copies will be destroyed.

Human use

Proposals selected for contract award are required to comply with provisions of the Common Rule (32 CFR 219) on the protection of human subjects in research (<http://www.dtic.mil/biosys/downloads/32cfr219.pdf>) and the Department of Defense Directive 3216.2 (<http://www.dtic.mil/whs/directives/corres/html2/d32162x.htm>). All proposals that involve the use of human subjects are required to include documentation of their ability to follow Federal guidelines for the protection of human subjects. This includes, but is not limited to, protocol approval mechanisms, approved Institutional Review Boards, and Federal Wide Assurances. These requirements are based on expected human use issues sometime during the entire length of the proposed effort.

For proposals involving “greater than minimal risk” to human subjects within the first year of the project, performers must provide evidence of protocol submission to a federally approved IRB at the time of final proposal submission to DARPA. For proposals that are forecasted to involve “greater than minimal risk” after the first year, a discussion on how and when the offeror will comply with submission to a federally approved IRB needs to be provided in the submission. More information on applicable federal regulations can be found at the Department of Health and Human Services – Office of Human Research Protections website (<http://www.dhhs.gov/ohrp/>).

Any aspects of a proposal involving human use should be specifically called out as a separate element of the statement of work and cost proposal to allow for independent review and approval of those elements. This applies to the Curriculum Team.

Security classification

Security classification guidance on a DD Form 254 (DoD Contract Security Classification Specification) will not be provided at this time since DARPA is soliciting ideas only. DARPA does not anticipate that any aspect of this program will be classified, and does not encourage classified proposals in response to this announcement. However, after reviewing incoming proposals, if a determination is made that contract award may result in access to classified information, a DD Form 254 will be issued upon contract award. ***If you choose to submit a classified proposal you must first receive the permission of the Original Classification Authority to use their information in replying to this announcement.***

Publication approval

DARPA has determined that the scope of the work for this program is contracted fundamental research. Therefore, public release of information about research performed on-campus at a university for this program is not subject to prior Government review.

The definition of Contracted Fundamental Research is contained in DOD Instruction 5230.27 and can be found at <http://www.dtic.mil/whs/directives/corres/pdf2/i523027p.pdf>. Public release of information about research performed under circumstances other than those described above is subject to prior government review, according to the procedures available

at <http://www.darpa.mil/tio>. Prime and subcontracts shall include DFARS clause 252.204-7000, Disclosure of Information.

Export Licenses

The Contractor shall comply with all U. S. export control laws and regulations, including the International Traffic in Arms Regulations (ITAR), 22 CFR Parts 120 through 130, and the Export Administration Regulations (EAR), 15 CFR Parts 730 through 799, in the performance of a resulting contract. In the absence of available license exemptions/exceptions, the Contractor shall be responsible for obtaining the appropriate licenses or other approvals, if required, for exports of hardware, technical data, and software, and for the provision of technical assistance.

The Contractor shall be responsible for obtaining export licenses, if required, before utilizing foreign persons in the performance of this contract, including instances where the work is to be performed on-site at any Government installation, including installations within the United States, where the foreign person will have access to export-controlled technical data or software.

The Contractor shall be responsible for all regulatory record keeping requirements associated with the use of licenses and license exemptions/exceptions.

The Contractor shall be responsible for ensuring that the provisions of this clause apply to its subcontractors.

CHAPTER 5. SUBMISSION PROCESS

Proposals not meeting the format described in this pamphlet may not be reviewed. (see Exception note below) Proposals **MUST** be submitted to DARPA in hard copy. Any submissions sent via fax or email will be disregarded. Responding to this announcement requires completion of an online Cover Sheet for each Proposal prior to submission. To do so, the offeror must go to <https://csc-ballston.dmeid.org/baa/index.asp?BAAid=07-04> and follow the instructions there.

Each offeror is responsible for printing the Confirmation Sheet and attaching it to every proposal copy. If an offeror intends to submit more than one Proposal, a unique UserId and password must be used in creating each Cover Sheet.

All proposals must include the following:

- One (1) print original of the full proposal including the Confirmation Sheet. Please do not use 3-ring binders.
- Three (3) hard copies of the full proposal including the Confirmation Sheet. Please do not use 3-ring binders.
- One (1) electronic copy of the full proposal. This electronic copy must be:
 - On a CD
 - In PDF or Microsoft Word for IBM-compatible format
 - clearly labeled with BAA 07-04, offeror organization, proposal title (short title recommended)

DARPA will acknowledge receipt of complete submissions and assign control numbers that should be used in all further correspondence regarding proposals.

The full proposal (original and designated number of hard and electronic copies) must be submitted in time to reach DARPA by 12:00 PM (ET) 18 January 2007 (initial closing), in order to be considered during the initial evaluation phase. However, BAA 07-04 BL will remain open until 12:00 NOON (ET) 14 November 2007 (final closing date). Thus, proposals may technically be submitted at any time from issuance of this announcement through 12:00 NOON (ET) 14 November 2007. Although proposals may be submitted at any time from issuance of this announcement through 12:00 NOON (ET) 14 November 2007, offerors are warned that the likelihood of funding is greatly reduced for proposals submitted after the initial closing date deadline.

Failure to comply with the submission procedures may result in the submission not being evaluated.

Exception: University (prime) grant submissions may be made via the Grants.gov web site, <http://www.grants.gov/>, by using the "Apply for Grants" function. Duplicate submissions following the above instructions are not required.

CHAPTER 6. REPORTING REQUIREMENTS/PROCEDURES:

The award document for each proposal selected and funded will contain a mandatory requirement for four DARPA/IPTO Quarterly Status Reports each year, one of which will be an annual project summary. These reports will be electronically submitted by each awardee under this BAA via the DARPA/IPTO Technical – Financial Information Management System (T-FIMS). The T-FIMS URL and instructions will be furnished by the contracting agent upon award.

There may also be additional reporting requirements for grants and cooperative agreements. DARPA's Contracts Management Office (CMO) website (<http://www.darpa.mil/cmo/pages/modelgrantagreement.htm>) contains information about model grant and cooperative agreement terms and conditions. Patent and invention reporting will be made on form DD882 or via iEdison electronic reporting.

CHAPTER 7. PROPOSAL PREPARATION AND FORMAT

The proposal shall be delivered in two volumes, Volume 1 (technical proposal) and Volume 2 (cost proposal). The technical volume should include sections I, II, and optionally III as described below. The cost volume should include section IV as described below.

Proposals shall include the following sections, each starting on a new page (where a "page" is 8-1/2 by 11 inches with type not smaller than 12 point) and with text on one side only. Apart from what is described in Section III, the submission of other supporting materials along with the proposal is strongly discouraged. All submissions must be in English.

Individual elements of Sections I and II of the proposal shall not exceed the total of the maximum page lengths for each section as shown in braces { } below.

Section I. Administrative

Confirmation Sheet

The confirmation sheet (described in Chapter 5 - Submission Process) will contain the following information:

- Announcement number;
- Technical topic area (Learning or Curricula)
- Proposal title;
- Technical point of contact including: name, telephone number, electronic mail address, fax (if available) and mailing address;
- Administrative point of contact including: name, telephone number, electronic mail address, fax (if available) and mailing address;
- Summary of the costs of the proposed research, including total base cost, estimates of base cost in each year of the effort, estimates of itemized options in each year of the effort, and cost sharing if relevant;
- Contractor's type of business, selected from among the following categories: "WOMEN-OWNED LARGE BUSINESS," "OTHER LARGE BUSINESS," "SMALL DISADVANTAGED BUSINESS [Identify ethnic group from among the following: Asian-Indian American, Asian-Pacific American, Black American, Hispanic American, Native American, or Other]," "WOMEN-OWNED SMALL BUSINESS," "OTHER SMALL BUSINESS," "HBCU," "MI," "OTHER EDUCATIONAL," "OTHER NONPROFIT", or "FOREIGN CONCERN/ENTITY.";

Section II. Detailed Proposal Information

This section provides the detailed discussion of the proposed work necessary to enable an in-depth review of the specific technical and managerial issues. Specific attention must be given to addressing both risk and payoff of the proposed work that make it desirable to DARPA. Page counts listed in braces are maximums.

A. Innovative claims for the proposed research {1 Page}: This page is the centerpiece of the proposal and should succinctly describe the unique approach taken to the entire “electronic student” or “curriculum development” problem.

B. Proposal Roadmap {1 Page}: The roadmap provides a top-level view of the content and structure of the proposal. It contains a synopsis (or "sound bite") for each of the nine areas defined below. It is important to make the synopses as explicit and informative as possible. The roadmap must also cross-reference the proposal page number(s) where each area is elaborated. The nine roadmap areas are:

- Main goals of the proposed research (stated in terms of new, operational capabilities for assuring that critical information is available to key users).
- Tangible benefits to end users (i.e., benefits of the capabilities afforded if the proposed technology is successful).
- Critical technical barriers (i.e., technical limitations that have, in the past, prevented achieving the proposed results).
- Main elements of the proposed approach.
- Rationale that builds confidence that the proposed approach will overcome the technical barriers. ("We have a good team and good technology" is not a useful statement).
- Nature of expected results (unique/innovative/critical capabilities to result from this effort, and form in which they will be defined).
- Criteria for scientifically evaluating progress and capabilities on an annual basis.
- Cost of the proposed effort for each performance year.

C. Research Objectives {2 Pages}

- *Problem Description.* Provide concise description of the problem area addressed by this research project, and how it differs from or refines the stated problem of the BLP.
- *Research Goals.* Identify specific research goals of this project. Identify and quantify expected performance improvements from this research. Identify new capabilities enabled by this research. Identify and discuss salient features and capabilities of developmental software prototypes.
- *Expected Impact.* Describe expected impact of the research project, if successful, to problem area.

D. Technical Approach {15 Pages}: Provide detailed description of technical approach that will be used in this project to achieve research goals.

- **Curriculum team candidates should provide the following:**
 - A description of the proposed innovative approach to semi-automating the creation of different types of curriculum materials corresponding to different NI methods

- A description of the proposed innovative approach to the rapid creation of curriculum material generators
- A description of the proposed innovative approach to the cost-effective creation of sufficient electronic teacher functionality to support interactive NI-methods
- For each curriculum domain and target task:
 - Explain its military relevance (if any) and describe the transitions for which the associated curriculum prepares the program
 - Describe the proposed curriculum in terms of the instructional scenario and the specific associated performance goal
 - Describe the simulated world that would be used and the potential reuse of existing relevant technology
 - List and describe any preexisting relevant tutoring systems and training materials, as well as the associated level and cost of potential reuse
 - Indicate any domains that are particularly well-suited for human comparison and explain why
- Mitigation report and plan for potential conflict of interest
- **Learning team candidates should provide the following:**
 - The proposed innovative approach for the top level controller (i.e., integration architecture) of the learning system.
 - A description of the set of NI methods that are expected to be supported, as well as interesting approaches for addressing specific interaction modalities (e.g., linguistic, etc.), that provide innovative leverage for addressing one or more NI method implementations.
 - The proposed innovated approach for the integration of ancillary capabilities that bridge the natural instructional interactions and NI-methods. (In Appendix C DARPA has indicated introspective methods and limited inference as two ancillary capabilities, though others are possibly needed).
 - A description of innovative algorithmic approaches/capabilities organized around individual or groups of proposed NI-methods.
 - A section describing the capability(ies) provided by each subcontractor along with an explanation of her/his proposed innovative approaches. (It is anticipated that each subcontractor will provide one or more broad capabilities that will be used in combinatorial fashion to address the space of NI methods. Each contributor is expected to be a recognized leader in the specific area of contribution).

E. Comparison with Current Technology. {2 Pages} Describe state-of-the-art approaches and the limitations within the context of the problem area addressed by this research.

F. Statement of Work (SOW) {3 Pages} Write the SOW in plain English, outlining the scope of the effort and citing specific tasks to be performed. Also, include references to specific subcontractors if applicable, and specific contractor requirements.

G. Detailed Individual Effort Descriptions. {3 Pages} Provide detailed task descriptions for each individual effort and/or subcontractor in the schedule graphic. Articulate competencies that each adds to the learning or curriculum team.

H. Schedule and Milestones {1 Page} Provide an annotated graphic representation of project schedule including detail down to the major task or subcontractor level. This should include but not be limited to, a multi-phase development plan, which demonstrates a clear understanding of the proposed research; and a plan for periodic and increasingly robust experiments over the project life that will show applicability to the overall program concept. Show all project milestones. Use absolute designations for all dates. It is anticipated that most NI methods will be implemented early in the program, and that NI method contracts will be relaxed over time (see discussion of *relaxation trajectory* in Appendix C), as the work progresses. Delivery of NI methods and major points of relaxation should be indicated in the schedule.

I. Deliverables Description. {3 Pages} List all proposed deliverables, providing a detailed description for each one. Include in this section all proprietary claims to results, prototypes, or systems supporting and/or necessary for the use of the research, results, and/or prototype. If there are no proprietary claims, this should be stated. The offeror must submit a separate list of all technical data or computer software that will be furnished to the Government with other than unlimited rights (see DFARS 227.) Furthermore, the additional licensing agreements as discussed in the Intellectual Property section should be used to specify which licenses will apply to each component of the deliverables. Specify receiving organization and expected delivery date for each deliverable.

Learning Team Deliverables

The Learning Team will provide an end-to-end bootstrapped learner (i.e., Learning System) that achieves the levels of performance prescribed in the “Formal Evaluations of Learning Systems” section. The Learning System will take the form of two integrated deliverables (please see Chapter 3 for details).

- **Learning Components** – This deliverable comprises a set of learning components conforming to a standard that includes an agreed upon interface, the interaction language, and an agreed upon set of NI methods. DARPA, however, recognizes that these learning processes will be intrinsically different, often involving idiosyncratic controls. Thus, the learning process standard will not include restrictions on specific process controls. In general, the learning process standard is intended to facilitate, but not necessarily automate, transferring components to other learning teams.
- **Integration Architecture** – This deliverable is a top-level controller that reasons over NI inputs and invokes relevant learning processes in response.

- In addition, participation in all discussions and workshops to specify the form of the various types of genetic knowledge.

Curriculum Team Deliverables

The Curriculum Team will produce three kinds of deliverables: **curricula, genetic knowledge, and empirical findings**. The curriculum team's primary responsibilities in the BLP are (a) to oversee the creation and maintenance of the genetic knowledge specifications and implementations necessary for a learning system to be capable of learning in any domain, including final versions of the interaction language, the Interlingua, NI method contracts, and a specification and implementation of the complete testing framework, (b) to generate curricula sufficient for teaching a learning system the stated performance tasks in the selected domains, and (c) to conduct the formal evaluations of progress at the end of each program phase. The associated deliverables are detailed here:

- Curricula – Each phase, the curriculum team will deliver a set of curricula sufficiently diverse for testing the coverage of the BL systems over the range implied by the NI method contracts. Each curriculum package will consist of the following (please see Chapter 3 for details):
 - A world simulator that provides an abstracted representation of the world relevant to the task domain, in which the learning system (and perhaps the instructor) can perform actions and observe effects of those actions (e.g., practice a task).
 - Injected knowledge – the required starting knowledge for each curriculum has both shared and private components corresponding respectively to:
 - Common knowledge and capabilities that apply to many or all curricula. This would include high-level general ontologies (e.g., those use for introspective reasoning about processes), general reasoning mechanisms (e.g., a spatial reasoner), and other general purpose knowledge that provides a basis for Bootstrapped Learning.
 - Curriculum-specific knowledge required for individual curricula. This category is comprised of domain-specific models of the effects of actions within a domain, domain-specific algorithms, and other relevant capabilities.
 - A ladder curriculum
 - For each concept rung, appropriate scaffolding is provided along with:
 - A fixed set of starting conditions for each instructional experience (predefined or generated)
 - Instructional experience (provided by the instructor or generated in response to the student or world)
 - Implementation of an “electronic instructor” for the lesson (as suggested in Chapter 3, each instructor implementation may be reused for many lessons).
 - A formal articulation of the learning objective and a comparison of the learning process state with the goal state (optional)

- A specification or index of graduate-level performance (optional)
 - A solution for each concept rung expressed in the Interlingua. (This can be used by the BL system when it fails to learn a given rung, ensuring the availability of prerequisite knowledge for subsequent rungs.)
- Genetic knowledge
- The curriculum team is the steward of all genetic knowledge, including the three types listed below:
 - The interaction language specification
 - The Interlingua specification
 - The series of NI method contracts used in successive years as requested by DARPA and the learning teams. (Each NI method contract specifies parameters that can be used to adjust the restrictiveness of assumptions upon which the learning teams rely. Over time, these parameters are adjusted according to a relaxation trajectory such that learning systems must become increasingly independent.)
 - A formal specification of all genetic materials and their ongoing maintenance
 - The implementation and maintenance of infrastructure to support the genetic languages
 - Periodic (see Chapter 2) written assessments of the validity and utility of the current NI method contracts by a qualified expert (e.g., educational psychologist). See the sections pertaining to performer roles in Chapter 3 for details.
 - An architectural specification and implementation of all BLCs and all shared knowledge required for all curricula.
 - At least one example lesson or “vignette” for each new NI method that serves as a stable working example of the instructor-based execution of that NI method.
- Empirical Findings – the curriculum team also serves as the “evaluation team”, providing the evaluation results and analysis (please see Chapter 3 for details):
- Periodic results and analysis of learning system evaluations (many lessons include a defined performance task and measurement criterion for assessment)
 - In phases 2 and 3, results of empirical analysis of human learning on same materials, and comparative evaluation of learning system as specified in Chapter 3
- The maintenance and availability of a current downloadable BL platform that includes curricula and learning systems representing the best functional examples in their respective classes. This system is suitable for use by learning algorithm developers to test their code empirically. Thus, it is instrumented in ways specifically designed to facilitate empirical testing (e.g., controllable specification of scaffolding relaxation parameters, automatic aggregation of cross-validated results across factorial combination of option settings, etc.)

J. Technology Transition and Technology Transfer Plans. {2 Pages} (Note: Curriculum teams only) Please see Chapter 8 for relevant evaluation criteria.

Discuss plans for technology transition and transfer. Identify specific military and commercial organizations for technology transition or transfer. Specify anticipated dates for transition or transfer.

For each ladder curriculum proposed by the curriculum team, describe the relevance to military transition in two senses:

- Military relevance - how is the domain relevant to the military?
- Military transition - how does the domain lend itself to a transition (i.e., if the ladder curriculum is replaced with sensors, is the result an instructable military system, such as a trainable UAV)?

Identify intellectual property dependencies and proprietary limitations that would affect the proposed transition.

K. Personnel and Qualifications. {3 Pages} List of key personnel, concise summary of their qualifications, and discussion of proposer's previous accomplishments and work in this or closely related research areas. Indicate the level of effort to be expended by each person during each contract year and other (current and proposed) major sources of support for them and/or commitments of their efforts. DARPA expects all key personnel associated with a proposal to make substantial time commitment to the proposed activity.

- Learning Team – each capability provided by the learning team is expected to be delivered by a recognized leader in that field. Evidence of this should be provided.
- Curriculum Team – the curriculum team will be the steward of important languages, architectures, etc. for the BLP. Therefore, personnel leading the collaborative efforts on defining and refining these specifications should have recognized leadership experience in such a capacity, which should be documented in the proposal.

L. Facilities. {1 Page} Description of the facilities that would be used for the proposed effort. If any portion of the research is predicated upon the use of Government Owned Resources of any type, the offeror shall specifically identify the property or other resource required, the date the property or resource is required, the duration of the requirement, the source from which the resource is required, if known, and the impact on the research if the resource cannot be provided. If no Government Furnished Property is required for conduct of the proposed research, the proposal shall so state.

Curriculum team should state where formal BL system testing will take place, and where human comparison testing will occur. Note section on Human Use in Chapter 4.

M. Experimentation Plans. {1 Page} (Please see Chapter 3 for details regarding the empirical aspects of the program.) Offerors should identify experiments to test the hypotheses of their approaches and be willing to work with other contractors in order to develop joint experiments in a common testbed environment. Offerors should expect to participate in teams and workshops to provide specific technical background information to DARPA, attend semi-

annual Principal Investigator (PI) meetings, and participate in numerous other coordination meetings via teleconference or Video Teleconference (VTC). Funding to support these various group experimentation efforts should be included in technology project bids.

N. Organizational Conflict of Interest: Awards made under this BAA may be subject to the provisions of the Federal Acquisition Regulation (FAR) Subpart 9.5, Organizational Conflict of Interest. All offerors and proposed subcontractors must affirmatively state whether they are supporting any DARPA technical office(s) through an active contract or subcontract. All affirmations must state which office(s) the offeror supports, and identify the prime contract number. Affirmations should be furnished at the time of proposal submission. All facts relevant to the existence or potential existence of organizational conflicts of interest, as that term is defined in FAR 2.101, must be disclosed, organized by task and year. This disclosure shall include a description of the action the offeror has taken, or proposes to take, to avoid, neutralize, or mitigate such conflict.

O. Intellectual Property.

1. FARS/DFARS Noncommercial Items IP Restrictions: (Technical Data and Computer Software)

Proposers responding to this solicitation requesting a procurement contract to be issued under the FAR/DFARS, shall identify all noncommercial technical data, and noncommercial computer software that it plans to generate, develop, and/or deliver under any proposed award instrument in which the Government will acquire less than unlimited rights, and to assert specific restrictions on those deliverables. Proposers shall follow the format under DFARS 252.227-7017 for this stated purpose. In the event that proposers do not submit the list, the Government will assume that it automatically has “unlimited rights” to all noncommercial technical data, and noncommercial computer software generated, developed, and/or delivered under any award instrument, unless it is substantiated that development of the noncommercial technical data, and noncommercial computer software occurred with mixed funding. If mixed funding is anticipated in the development of noncommercial technical data, and noncommercial computer software generated, developed, and/or delivered under any award instrument, then proposers should identify the data, documentation, and software in question, as subject to Government Purpose Rights (GPR). In accordance with DFARS 252.227-7013 Rights in Technical Data - Noncommercial Items, and DFARS 252.227-7014 Rights in Noncommercial Computer Software and Noncommercial Computer Software Documentation, the Government will automatically assume that any such GPR restriction is limited to a period of five (5) years in accordance with the applicable DFARS clauses, at which time the Government will acquire “unlimited rights” unless the parties agree otherwise. PROPOSERS ARE ADVISED THAT OFFERS CONTAINING RESTRICTIONS ON INTELLECTUAL PROPERTY ARE BY NATURE LESS FAVORABLE AND VALUABLE TO THE GOVERNMENT. RESTRICTIONS WILL BE CONSIDERED IN THE EVALUATION PROCESS. If no restrictions are intended, then the proposer should state “NONE.”

A sample list for complying with this request is as follows:

NONCOMMERCIAL				
Technical Data Computer Software To be Furnished With Restrictions	Basis for Assertion	Asserted Rights Category	Name of Person Asserting Restrictions	Licensing Model
(LIST)	(LIST)	(LIST)	(LIST)	(LIST)

In the two sample lists (corresponding to noncommercial and commercial), the column titled “Licensing Model” should specify which of the additional licensing models proposed in the context of the relevant section below applies to each deliverable.

2. FARS/DFARS Commercial Items IP Restrictions: (Technical Data and Computer Software)

Proposers responding to this solicitation requesting a procurement contract to be issued under the FAR/DFARS, shall identify all commercial technical data, and commercial computer software that may be embedded in any noncommercial deliverables contemplated under the research effort, along with any applicable restrictions on the Government’s use of such commercial technical data and/or commercial computer software. In the event that proposers do not submit the list, the Government will assume that there are no restrictions on the Government’s use of such commercial items. PROPOSERS ARE ADVISED THAT OFFERS CONTAINING RESTRICTIONS ON INTELLECTUAL PROPERTY ARE BY NATURE LESS FAVORABLE AND VALUABLE TO THE GOVERNMENT. RESTRICTIONS WILL BE CONSIDERED IN THE EVALUATION PROCESS. If no restrictions are intended, then the proposer should state “NONE.”

A sample list for complying with this request is as follows:

COMMERCIAL				
Technical Data Computer Software To be Furnished With Restrictions	Basis for Assertion	Asserted Rights Category	Name of Person Asserting Restrictions	Licensing Model
(LIST)	(LIST)	(LIST)	(LIST)	(LIST)

3. Non-FARS/DFARS IP restrictions: (Technical Data and Computer Software)

Proposers responding to this solicitation requesting a Grant, Cooperative Agreement, Technology Investment Agreement, or Other Transaction for Prototype shall follow the applicable rules and regulations governing these various award instruments, but in all cases

should appropriately identify any potential restrictions on the Government's use of any Intellectual Property contemplated under those award instruments in question. This includes both Noncommercial Items and Commercial Items. Although not required, proposers may use a format similar to that described in Paragraphs 3.4.1 and 3.4.2 herein. **PROPOSERS ARE ADVISED THAT OFFERS CONTAINING RESTRICTIONS ON INTELLECTUAL PROPERTY ARE BY NATURE LESS FAVORABLE AND VALUABLE TO THE GOVERNMENT. RESTRICTIONS WILL BE CONSIDERED IN THE EVALUATION PROCESS.** If no restrictions are intended, then the proposer should state "NONE."

4. Patent Dependencies

Please include documentation proving your ownership of or possession of appropriate licensing rights to all patented inventions (or inventions for which a patent application has been filed) that will be utilized under your proposal for the DARPA program. If a patent application has been filed for an invention that your proposal utilizes, but the application has not yet been made publicly available and contains proprietary information, you may provide only the patent number, inventor name(s), assignee names (if any), filing date, filing date of any related provisional application, and a summary of the patent title, together with either: 1) a representation that you own the invention, or 2) proof of possession of appropriate licensing rights in the invention.

5. IP Representation

Please also provide a good faith representation that you either own or possess appropriate licensing rights to all other intellectual property that will be utilized under your proposal for the DARPA program. If you are unable to make such a representation concerning non-patent related intellectual property, please provide a listing of the intellectual property to which you do not have needed rights, and provide a detailed explanation concerning how and when you plan to obtain these rights.

IMPORTANT NOTE: IF THE OFFEROR DOES NOT COMPLY WITH THE ABOVE STATED REQUIREMENTS, THE PROPOSAL MAY BE REJECTED.

P. Licensing Model {5 Pages}: In addition to the conventional licensing as listed above, this program seeks to extend and distribute materials developed in this program in order to further the science of Bootstrapped Learning. Therefore, this section has been included to request proposals for specific broader licensing terms that would be acceptable to the performers. Please document clearly the licensing model, explain how it will contribute to the licensing goals of the program as stated below, and provide one or more sample licensing agreements.

Goals

A central goal of the BL program is to leave the legacy of a publicly available BL toolkit that will permit the Machine Learning community, and indeed, the Artificial Intelligence community at large, to use and extend the BL framework and research collateral. Ideally, this toolkit would be freely downloadable and contain everything necessary for any researcher to

conduct BL research, including the development and testing of new learning algorithms and generation of new curricula.

Thus, licenses should be constructed to minimize barriers for the following primary use cases:

- Ongoing extension and reuse of delivered curricula
- Ongoing extension and reuse of the entire delivered bootstrapped learning systems
- Ongoing extension and reuse of individual learning algorithm components

Guidelines

The following use restriction scale is based upon divisions that are considered relevant to this program. Therefore, this constitutes an appropriate metric for use in the section of the proposal that describes the license(s), and will also be used by the Government when evaluating your plan. Please note that this scale is cumulative (e.g., #2 subsumes #1, #3 subsumes both #2 and #1, etc.). Thus, higher numbers are less restrictive.

- Non-commercial (research-only) license – for non-commercial academic use; free or reasonably-priced
- Non-commercial license – includes source code, which may be modified for internal use
- Non-commercial license – modified source code may be redistributed – permits the distribution of modified source code and executables derived from modified source
- Unlimited usage license (e.g., commercial use) – no restrictions on usage or distribution

When addressing the use cases listed above, please indicate licensing considerations for the following:

- Execution support technologies (e.g., operating systems, runtime libraries, etc.)
- Development support technologies (e.g., editors, proprietary tools used to develop program technologies, etc.)

A significant criterion for proposal evaluation will be the intellectual property licensing model. In general, **less restrictive licenses are considered favorable**. Since the research value of the set of deliverables depends upon the collective availability of its constituents, **licensing models will be evaluated primarily in terms of their most restrictive provisions** in the context of program goals.

As indicated previously, a key objective in this program is to promote the use and extension of BL technologies. This dictum cannot be overstated. Therefore, any licensing restrictions that introduce barriers to this objective would be viewed negatively. If the proposed licensing model does not comply, the proposer must recommend an amelioration strategy (e.g., open source alternatives, research versions, etc.) be provided.

Example

The following matrix depicts an example licensing model that is consistent with program goals. Each cell indicates one or more licenses that represent the intended level of restriction. The Open Source Initiative provides a formal definition of *open source* on their website at:

<http://www.opensource.org/docs/definition.php>

Licenses that fit this definition are represented in the matrix below as open source licenses or “OSL”. Examples of conforming licenses are listed here:

<http://www.opensource.org/licenses/>

In addition, the Java Research License is available online at <http://java.net/jrl.html>. Note: OS = Operating System, and RTL = Run-time Library.

	Curriculum	Controller	Learning Algorithms
Program Technology	Curricula: OSL Pre-existing proprietary simulator: JRL	OSL	OSL
Execution Support	OS and RTL: OEM license All other: JRL	OS and RTL: OEM license All other: JRL	OS and RTL: OEM license All other: JRL
Development Support	JRL	JRL	JRL

Section III. Additional Technical Information

Reprints or copies of **up to three** relevant technical papers and research notes (published and unpublished) that document the technical ideas, upon which the proposal is based, may be included in the proposal submission. Provide two hard copies and a soft copy. Please note: The materials provided in this section, and submitted with the proposal, will be considered for the reviewer’s convenience only and not considered as part of the proposal for evaluation purposes.

Section IV. Cost proposal

The cost volume should be a separate document from the technical and management volume comprising sections I through III.

A. Cover sheet

- Name and address of offeror (include zip code);
- Name, title, and telephone number of offeror's point of contact;
- Award instrument requested: cost-plus-fixed-fee (CPFF), cost-contract--no fee, cost sharing contract--no fee, or other type of procurement contract (specify), grant, agreement, or other award instrument;
- Place(s) and period(s) of performance;
- Funds requested from DARPA for the Base Effort, each option and the total proposed cost; and the amount of cost share (if any);
- Name, mailing address, telephone number and Point of Contact of the offerors cognizant government administration office (i.e., Office of Naval Research/Defense Contract Management Agency (DCMA)) (if known);
- Name, mailing address, telephone number, and Point of Contact of the Offeror's cognizant Defense Contract Audit Agency (DCAA) audit office (if known);
- Any Forward Pricing Rate Agreement, other such Approved Rate Information, or such other documentation that may assist in expediting negotiations (if available);
- Contractor and Government Entity (CAGE) Code,
- Dun and Bradstreet (DUN) Number;
- North American Industrial Classification System (NAICS) Number [NOTE: This was formerly the Standard Industrial Classification (SIC) Number]; and,
- Taxpayer Identification Number (TIN).
- All subcontractor proposal backup documentation to include items a. through l. above, as is applicable and available).

B. Detailed cost breakdown

Provide total program cost for core work by phase, with Phase 2 and Phase 3 shown as options, e.g. Base (Phase 1) plus Option 1 (Phase 2) plus Option 2 (Phase 3.)

Curriculum Teams Only: Provide a further breakdown of costs as shown in the tables below. The core costs represent the costs of the infrastructure required for the curriculum team to provide evaluation, support of the genetic knowledge, etc. Curriculum domains are costed as options to be executed over the three phases. New or extended curricula ladders within each option will be delivered in each of the phases of the program. Details about the domains can

be found in Chapter 3-Performer Requirements. Each curriculum domain option should provide costing in the first phase for the simulation platform and at least one curriculum. In subsequent phases these options should list the cost of extending the curricula delivered in phase one and/or creation of one or more additional curricula with in the same domain. The hidden curriculum ladders (see Chapter 3 section on Curriculum Ladders) should be listed as a single cost option covering the simulator and a single curriculum.

We include a table below to outline the format for these costing options:

Curriculum Domain Costs	Phase 1	Phase 2	Phase 3
Core Costs	\$	\$	\$
Domain Option 1	\$	\$	\$
Domain Option 2	\$	\$	\$
Domain Option 3	\$	\$	\$
Domain Option 4	\$	\$	\$
...			

Costs for Hidden Domains	
Hidden Domain 1	\$
Hidden Domain 2	\$
...	

Cost breakdown categories (for both learning and curriculum teams):

- Direct Labor – Individual labor category or person, with associated labor hours and unburdened direct labor rates;
- Indirect Costs – Fringe Benefits, Overhead, General and Administrative Expense, Cost of Money, etc. (Must show base amount and rate);
- Travel – Number of trips, number of days per trip, departure and arrival destinations, number of people, etc.

- Subcontract – A cost proposal as detailed as the offeror's cost proposal will be required to be submitted by the subcontractor. The subcontractor's cost proposal can be provided in a sealed envelope with the offeror's cost proposal or will be requested from the subcontractor at a later date;
- Consultant – Provide consultant agreement or other document which verifies the proposed loaded daily/hourly rate;
- Materials – Should be specifically itemized with costs or estimated costs. An explanation of any estimating factors, including their derivation and application, shall be provided. Please include a brief description of the offeror's procurement method to be used;
- Other Direct Costs – Should be itemized with costs or estimated costs. Backup documentation should be submitted to support proposed costs.
- Costs of major program tasks and major cost items by year and month;
- Supporting cost and pricing information.

Supplementary information should be provided in sufficient detail to substantiate the summary cost estimates above. Include a description of the method used to estimate costs and supporting documentation. Provide the basis of estimate for all proposed labor rates, indirect costs, overhead costs, other direct costs and materials, as applicable.

C. Government Furnished Property

Contractors requiring the purchase of information technology (IT) resources as Government Furnished Property (GFP) MUST attach to the submitted proposals the following information:

- A letter on corporate letterhead signed by a senior corporate official and addressed to Dr. Daniel Oblinger, Program Manager, DARPA/IPTO, stating that you either can not or will not provide the information technology (IT) resources necessary to conduct the said research.
- An explanation of the method of competitive acquisition or a sole source justification, as appropriate, for each IT resource item.
- If the resource is leased, a lease/purchase analysis clearly showing the reason for the lease decision.
- The cost for each IT resource item.

CHAPTER 8. EVALUATION AND FUNDING PROCESSES

It is the policy of DARPA to ensure impartial, equitable, comprehensive proposal evaluations and to select the source (or sources) whose offer meets the Government's technical, policy, and programmatic goals. Pursuant to FAR 35.016, the primary basis for selecting proposals for acceptance shall be technical, importance to agency programs, and fund availability. In order to provide the desired evaluation, qualified Government personnel will conduct reviews and (if necessary) convene panels of experts in the appropriate areas.

Evaluation of proposals will be accomplished through a scientific review of each proposal using the following criteria. While these criteria are listed in descending order of relative importance, it should be noted that the combination of all non-cost evaluation factors is significantly more important than cost.

Proposals will not be evaluated against each other, since they are not submitted in accordance with a common work statement. DARPA's intent is to review proposals as soon as possible after they arrive; however, proposals may be reviewed periodically for administrative reasons. For evaluation purposes, a proposal is the document described in Chapter 7, "PROPOSAL PREPARATION AND FORMAT," Section I, II and Section IV. Other supporting or background materials (Section III) submitted with the proposal will be considered for the reviewer's convenience only and not considered as part of the proposal.

Evaluation of proposals will be accomplished through a scientific review of each proposal using the following criteria, which are listed in descending order of relative importance:

- **Overall Scientific and Technical Merit:** The objective of this criterion is to establish the technical worthiness of the proposed effort. Evaluation will consider problem understanding, problem formulation, and the potential for long term technology consequences. The potential for revolutionary impact must be evident. The proposal should offer the potential to influence the long-term research agenda in relevant fields. The proposal should pursue theory with an eye toward eventual application. The problem formulation, technical obstacles, and theoretical enablers should be clearly and soundly articulated. The proposal should evidence awareness of both historical and ongoing related work. Validation of results should be considered. Risks should be appropriately identified, characterized, and mitigated. Furthermore, as a goal of the program is the advancement of science, proposals should maximize the availability and reuse of BL technologies to further work in this field. In particular, this criterion will assess the extent to which the IP stipulations of the proposal are consistent with the scientific goals of the program.
- **Innovative Technical Solution to the Problem:** The objective of this criterion is to establish that innovative and promising approaches are being applied to achieve the objectives of the effort. Offerors should apply new and/or existing theory and practice in an innovative way that supports the objectives of the proposed effort. The proposed approach concepts should show breadth of innovation across all the dimensions of the proposed solution. The theoretical enablers should be traceable to the objectives defined in the proposal.

- **Offeror's Capabilities and Related Experience:** The objective of this criterion is to establish that the offeror has credible capability and experience to complete the proposed work. The qualifications, capabilities, and demonstrated achievements of the proposed principals and other key personnel for the primary and subcontractor organizations must be clearly shown. Moreover, the key individuals must plan to commit sufficient time to the project to ensure its success. The proposers should have a track record of innovation and leadership in the relevant disciplines, and should be professionally well-positioned to influence the research agendas of entire disciplines. Proposers should have sufficient professional and research expertise to be able to react appropriately, plan, and re-plan when serendipitous technical advances and negative results arise.
- **Potential Contribution and Relevance to DARPA/IPTO Mission:** The objective of this criterion is to establish a strong link between this work and the DARPA/IPTO mission. It is NOT necessary that the proposed work be immediately usable in military systems. It is only necessary that this work contribute to technical areas of need by the DOD. Evaluation of this criterion will consider factors such as the likelihood of transitioning theory into networking practice, as opposed to evaluating the likelihood of transitioning systems into military practice. Also considered will be impediments to future transition, including intellectual property restrictions.
- **Plans and Capability to Accomplish Technology Transition:** Note: This criterion is only applicable to proposals from Curriculum Teams. The offeror should provide a clear explanation of how the technologies to be developed will be transitioned to capabilities for military forces. Technology transition should be a major consideration in the design of experiments, particularly considering the potential for involving potential transition organizations in the experimentation process.
- **Cost Realism:** The objective of this criterion is to evaluate whether the costs are aligned with the proposed work plan, whether strategies for cost reduction are being employed effectively, and whether the overall cost/benefit ratio is deemed appropriate. The overall estimated cost to accomplish the effort should be clearly shown as well as the substantiation of the costs for the technical complexity described. Evaluation will consider the value of the research to Government and the extent to which the proposed management plan will effectively allocate resources to achieve the capabilities proposed. Creative approaches to reduce costs by leveraging other ongoing research will be viewed favorably, particularly in support of experimentation. Overall cost is considered a substantial evaluation criterion but is secondary to technical excellence. Unrealistically low cost estimates are as undesirable as unreasonably high costs. In general, the proposal cost should be commensurate with the work effort proposed, adequate detail must be provided to allow proper evaluation of the cost rationale, and cost effective measures must be employed wherever possible.

As soon as the evaluation of a proposal is complete, the offeror will be notified that 1) the proposal has been selected for funding pending contract negotiations, or 2) the proposal has not been selected. Additionally, DARPA reserves the right to award without discussions, and to accept proposals in their entirety or to select only portions of proposals for award. In the

event that DARPA desires to award only portions of a proposal, negotiations may be opened with that offeror. The Government reserves the right to fund proposals in phases with options for continued work at the end of one or more of the phases. Awards under this BAA will be made to offerors on the basis of the evaluation criteria listed above, and program balance to provide best value to the Government.

The Government reserves the right to select all, some, or none of the proposals received in response to this solicitation and to make awards without discussions with offerors; however, the Government reserves the right to conduct discussions if the Source Selection Authority later determines them to be necessary. Proposals identified for funding may result in a contract, grant, cooperative agreement, or other transaction depending upon the nature of the work proposed, the required degree of interaction between parties, and other factors. If warranted, portions of resulting awards may be segregated into pre-priced options. The government reserves the right to choose the appropriate funding instrument.

CHAPTER 9. BAA CORRESPONDENCE AND ADMINISTRATIVE ADDRESSES

DARPA will use electronic mail for all technical and administrative correspondence regarding this BAA, with the exception of selected/not-selected notifications. These official notifications will be sent via US mail to the Technical POC identified on the proposal coversheet.

Administrative, technical or contractual questions should be sent via e-mail to BLSolicitation@darpa.mil. If e-mail is not available, please fax questions to (703) 741-7804, Attention: BL Solicitation. All requests must include the name, email address, and phone number of a point of contact.

Solicitation Web site and Electronic File Retrieval:
<http://www.darpa.mil/ipto/solicitations/solicitations.htm>.

Postal address: DARPA/IPTO, ATTN: BAA 07-04, 3701 N. Fairfax Drive, Fairfax, VA 22203-1714. For deliveries that require a phone number, such as FedEx or UPS, please use 703-696-2356, which is the DARPA mailroom. For hand deliveries, the courier shall deliver the package to the DARPA Visitor Control Center at the address specified above. To ensure proper handling, the outer package, as well as the cover page of the proposal, must be marked "IPTO BAA 07-04."

APPENDIX A: EXTERNAL MATERIALS

External resources relevant to the technical approach are available online at:

<http://www.sainc.com/bl-extmat/index.html>

This site is segregated into two main sections: BL framework materials, and BL examples. The framework materials section includes an overview of BL along with the framework language specifications, including the interaction language, Interlingua, and NI-methods considerations. The examples section contains several domain and NI method-specific examples of how BL might occur.

Use of Materials

All online materials are tagged with dates and updated periodically to reflect their current state of evolution. **Proposals should refer only to materials provided on or before the BAA opening date.** Versions of these materials as they existed when the BAA was released will be maintained and indicated on the website.

Please note that these external materials are not part of the BAA. Rather they are attempts by DARPA to clarify technical aspects of the program in order to speed its execution. In fact, DARPA is not committed to details they contain. Rather they are an attempt to illuminate technical approaches that could be employed throughout the course of the BLP.

Languages

Because initial versions of the Interaction Language and Interlingua will provide important touchstones for the BL program, DARPA is actively developing a first implementation of these languages prior to the start of the program.

Researchers may refer to these languages in their proposals, and are also free to propose alternatives or extensions as required by their respective methodologies. The initial versions of these languages are constantly evolving, therefore, an official version will be provided at the release of the BAA. This "official" version is the only one that should be referenced within proposals, if necessary. In order to keep the community informed of progress, additional changes may be posted as an understanding of these languages advances. Proposals should not refer to these later versions. In fact, all proposals will be understood in the context of the language versions that coincide with the BAA release.

APPENDIX B: BACKGROUND & MOTIVATION

Contrasting BL with Traditional ML

BL techniques provide the means to learn from a teacher in ways analogous to those used by human students. The teacher is assumed to have a priori knowledge and capabilities that the student is expected to learn. In contrast, traditional ML focuses primarily on learning capabilities not possessed by the teacher. Indeed, in ML there usually is not a teacher in the canonical sense.

The basic distinction between ML and BL can be characterized in terms of the learning agendas. ML manifests typically as an approach to knowledge *discovery*, whereas BL is a framework for knowledge *communication*. Table 2 compares BL and traditional ML along several dimensions, including this one.

Bootstrapped Learning	Traditional Machine Learning
Knowledge transfer	Knowledge discovery
Learn interactively, structure of learning guided by instructor	Self-directed learning
Requires small sets of instructor-annotated data	Requires larger sets of data
Knowledge-rich instructor that possesses performance capacity sought by the student	No instructor; instead, a user with a partial understanding of the target capability
“Laddered” curricula (lessons are structured to teach base knowledge first; later lessons build upon earlier lessons)	Unstructured “flat” learning of a target given some bias and some data

Table 2: Comparison of Bootstrapped Learning and Traditional Machine Learning

BL is a method for communicating capabilities to a naïve student from an instructor who is capable at a *graduate level* for the relevant curriculum.

[The generation of these curriculum-specific performance benchmarks is described in Chapter 3 under “Formal Evaluations of the Learning System.”] By contrast, ML is primarily a modeling approach that is used when something, *but not everything relevant* to some target problem is known.

A central goal of BL is to create *domain-independent learning processes* for each NI method. The emphasis on natural instruction results in substantially different research problems than

those investigated in traditional ML. Consider the class of NI methods, *instruction by examples*, that was introduced in Chapter 1. At first glance, this kind of NI method may seem very much like a supervised learning task. However, there are important differences: the NI method provides only as much training data as would be typically shown to a human student, usually on the order of a dozen or less examples. This is in contrast to the dozens or thousands of examples that would be used in ML. However, since *instruction by example* is based upon intentional instruction, one can assume that, in contrast to the random sampling that is typical of traditional ML, the example set is contrived by the instructor to facilitate incremental learning. Furthermore, because instructors are assumed to understand the concepts being learned, they can provide many forms of hints (scaffolding) that simplify learning. In general, any information that one human might provide to another during instruction is a candidate input for an NI method.

In addition to natural instruction, bootstrapping distinguishes BL from traditional ML by further reducing learning complexity. Early lessons (lower rungs) on the ladder curriculum are composed of small, highly constrained learning tasks. The instructor designs these lessons such that the knowledge gained through their successful completion provides a foundation for learning more complex lessons. Subsequent lessons, involving increased complexity and abstraction, are similarly built upon previous lessons. In this way, the learning system effectively climbs the ladder one rung at a time.

The BL approach constitutes a paradigm shift, and is a radical departure from traditional ML:

- The decomposition embodied by the interaction language provides for both domain and instruction method generality by isolating the learning processes. For the first time, learning processes must be dealt with as entirely autonomous processes without external configuration, rather than merely as specialized tools.
- Bootstrapping requires careful attention to the representation of the initial domain knowledge (e.g., the software BLCs). This representation then supports chaining indefinitely the outputs of one learning rung into the inputs of the next rung, within a given task domain.
- BL is truly domain-independent because researchers must construct the entire electronic student learning system without any knowledge of the target domain. In fact, in testing during later phases of the program, the learning system developers will not know the task domain until after the learning system is submitted for testing.
- A single, unmodified learning system can handle many different ladders without intervention, reprogramming or reconfiguration.
- A successful learning system supports mixed-initiative learning. The student can request which lessons are taught and the order in which they are presented. The teacher, which is embedded in the ladder, dictates which NI method(s) are available for teaching a given lesson. Furthermore, different NI-methods combine different forms of student and teacher-directed interaction. As a consequence of this arrangement, the learning system must be capable of accepting interleaving of instructional methods as dictated by the instructor's provided materials.

- The BLP is designed to isolate and address the most central research issues in BL, in particular, those directly relevant to learning. **The program explicitly sidesteps the issue of perception abstraction across all possible domains**, instead relying upon other DARPA programs that focus on specific perceptual interaction modalities (e.g., natural language processing, visual object recognition, etc.) and domains to address those challenges.
- A valuable byproduct of the BLP will be a self-contained test harness that is optimized for rapidly testing new learning algorithms simultaneously against many performance task domains.

In order to provide a vocabulary for discussing these differences we define the two applications of ML:

Data Modeling The application of ML to the problem of knowledge **discovery**—the problem of identifying a predictive model (or effective performance capability) given some presented data, and desired objective function.

Assumption: A model of the given data **is not** known.

The assumption underlying the use of ML in the modeling context is that ML has been invoked *precisely because* an effective model is not available. This is in contrast to this second application of ML:

Instructable Computing - The application of ML to the problem of knowledge **communication**—the problem of transmitting some model or capability from an instructor who has it to a student who does not.

Assumption: A model of the given data **is** known, but a mapping of that model from instructor to student **is not** known.

In this new context, what is unknown to both instructor and student is how to map model or capability known to the instructor onto the knowledge and capabilities known to the student. Thus, there is a space of choices in this new application of ML, but it is not choices about how to model the world, rather it is how to map knowledge between agents.

This is a difference that makes a large difference in how one peruses ML in these two contexts. In the traditional context one attempts to make few assumptions about the nature of the data source. A common statistical assumption for example is to assume the data is drawn IID (independent and identically-distributed) from a source distribution. In instructable computing both instructor and student understand the goal is knowledge transmission, and they both understand the instructor can perform the prediction or performance task, thus many very directed “hints” can be provided as part of the data in the instructable computing context that would be precluded by the assumptions of the data modeling context. We believe exploration of ML in this new “Instructable Computing” context will require significant

extensions to existing ML, and will herald a new usage class for ML—as a tool for instructing (programming/configuring) software systems.

Why Is There an Opportunity for BL Now?

Much research has been done in ML, especially during the past decade. Although BL has not been addressed directly, it is clear that much of the work in ML is relevant. In fact, it is hoped that many existing ML procedures will be adapted to operate within the BL framework. In addition, the educational and cognitive psychology communities have recently increased the level of scientific rigor used in evaluating teaching strategies and methods (e.g., work at the NSF-sponsored Pittsburgh Science of Learning Center, as one example among many). The methods they have already explicated, along with those they will in the near future, may inform this work by revealing promising new areas for which to develop new incremental learning processes. Additionally, computational systems are now being routinely fielded that have sufficient computational resources to support field instruction based upon the requirements of instructor-led NI methods.

Why is there confidence that the BL concept is feasible?

- BL is designed to permit the acquisition of complex structures, as provided explicitly by an instructor, that are outside the grasp of statistical ML.
- The BL paradigm reduces the learning search space exponentially, in three different ways (discussed below).
- Learning is isolated from problem specifics, which permits bootstrapping toward an arbitrary task.
- Learning processes are specialized to the type of NI methods employed rather than the idiosyncrasies of the problem domain. This allows them to be general, while at the same time providing them access to important characteristics of the instructional environment. Depending upon these characteristics yields strongly biased learning.

The search space of candidate knowledge representations suitable for explaining a curriculum ladder as large and complex as the ones contemplated for BL is expansive. However, the BL paradigm is expected to reduce this space to a practicable level by focusing on the relevant issues at each step of the learning process. As illustrated in Figure 7, BL significantly reduces the complexity of the learning problem in three different ways:

The structure of the curriculum ladder decomposes the enormous initial learning search space into a series of much smaller spaces, one for each rung of the ladder.

Within each rung, learning is further constrained by the rich inputs made possible by NI methods.

The structure of relevant BLCs constrains the size of the space over which the learning processes must search for appropriate solutions.

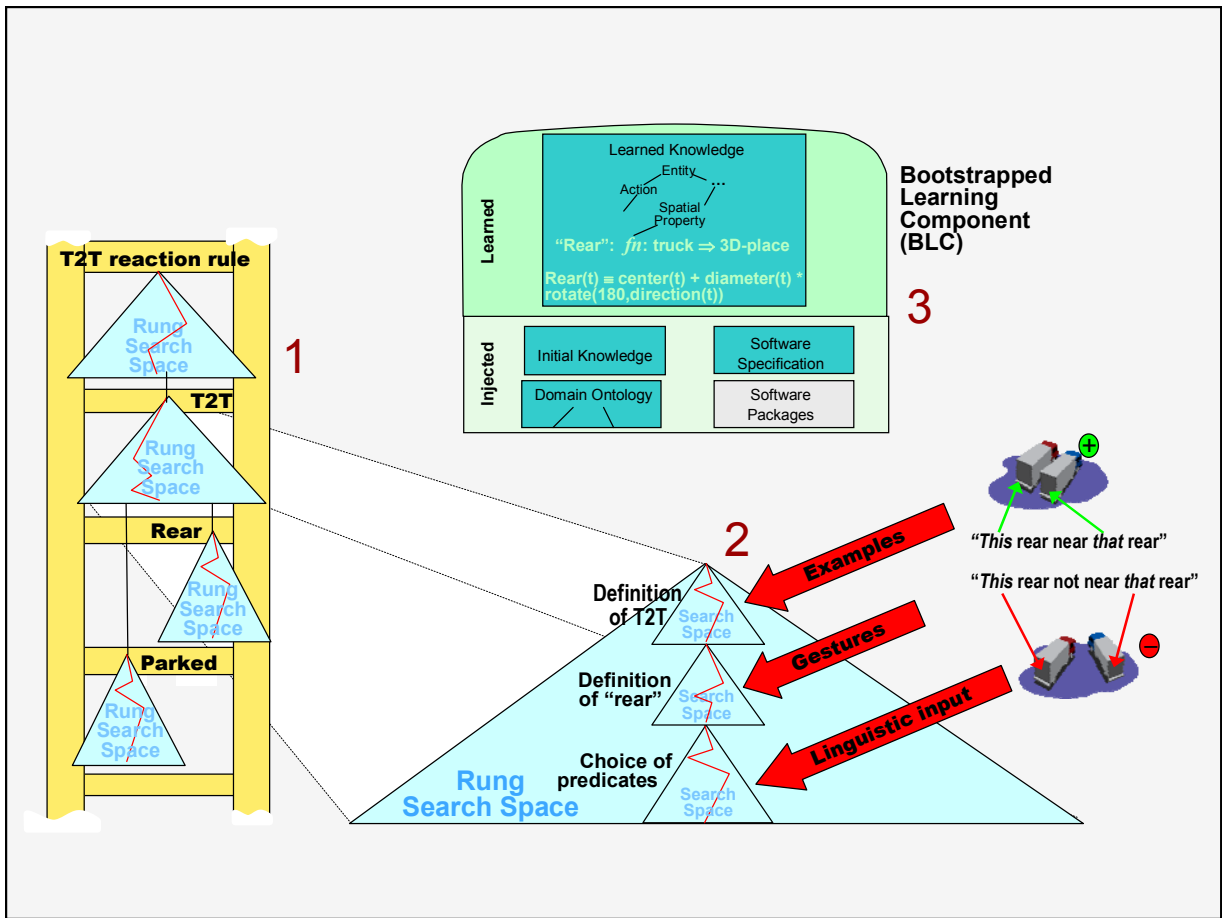


Figure 7: Natural Instruction Makes BL Feasible

BL Will Impact Multiple Research Communities

If the BLP is successful, it will impact the ML research community by greatly expanding the notion of what form automated learning can take and where the important future research avenues may lie. Progress on individual BL techniques, especially those that come initially from ML, will also enhance work in non-BL areas.

To help the shift toward research emphasis, BLP will provide sophisticated datasets to drive the research of the nascent “instruction-based learning” community. In addition, the test harness, described earlier, will allow individual researchers to develop and test new NI-based BL processes for the first time.

The BLP may also make a difference to the educational and cognitive psychology communities. The BLP curriculum can be used to teach humans as well as learning systems. Thus, it could prove useful as a test bed for tightly controlled teaching studies.

Perhaps the greatest influence of a successful BLP would be to the software engineering and program synthesis research communities. BL could lead to new paradigms involving

programming computers by high-level instruction. The new programming language would be based upon the computational substrate over which the learning system learns (e.g., the basic operations of an unmanned air vehicle [“UAV”]). In the new paradigm, however, the programmer would be replaced by the BL teacher and programming would be accomplished without complete knowledge of the computer programming language. Instead, the new, high-level languages would consist of “programming by example,” “programming by definition,” “programming by instruction,” etc.

BL Will Significantly Expand the Use of Computers

Encapsulated Trainability

The ultimate objective of the BLP is to create an “electronic student” endowed with human-like learning abilities that performs at a level comparable to humans. A compelling method for validating this involves comparing the learning system’s learning performance with typical human performance, using a common curriculum. Indeed, the program’s evaluation criterion (Go/No Go) will be based upon such a test. However, the BLP’s more immediate objective is that of creating a “reusable trainability wrapper” that can be used to instantly transform any existing system into one that is teachable using NI methods. Some ladders in the BLP will be geared specifically toward this objective.

The ability of humans to transfer knowledge (i.e., instruct) to a machine much as they would transfer knowledge to another human could have a profound effect on many AI and other software systems: it would allow ordinary people (i.e., non-AI experts) to enter knowledge into systems and alter system behavior. In the long run such a capability could obviate the need for a programmer when making simple to moderately complex functionality modifications.

Field Trainable Department of Defense (DoD) Systems

DoD systems would derive significant utility gains by becoming “field trainable.” In particular, systems in the areas of robotics and tactical command, control, intelligence, surveillance, and reconnaissance (C2ISR) would be extensively affected. Support areas, such as logistics, would also be affected.

The battlefield is a dynamic environment, whether it occurs as a large force-on-force engagement in open terrain or as a small unit pursuing a terrorist group in an urban environment. The enemy reacts to new adversarial capabilities in near real-time by adjusting its doctrine (e.g., how large a group it uses for a particular mission, how it is organized, and how it communicates) and tactics. Systems that track the enemy, predict movements, and provide response options cannot be preprogrammed to support every contingency. Thus, in order to retain their efficacy, they need to be updated when changes in the enemy’s strategies are detected. However, such changes usually require a time-consuming, costly software update cycle that cannot be implemented in the field and cannot be performed fast enough to keep pace with the enemy’s changes.

BL would enable a warfighter, who is the system operator and/or subject-matter expert, to modify the system's performance directly and in a timely fashion. For example, in response to intelligence indicating that the enemy has shifted from using trucks to using cars for transporting munitions, a UAV operator might wish to direct the UAV to search for different types of vehicles while preserving the operational strategies already in use.

Conceivably, *any* military hardware with a CPU, sensors, and actuators would be field trainable. Candidate applications of this technology include, for example, robotic systems, unmanned vehicles (air, ground, undersea), unattended sensors, surveillance systems, video understanding systems, situation assessment systems, summarization systems, simulation and training systems, and planning and scheduling systems.

Commercial and Other Applications

For every military application there are multiple analogous commercial applications, spanning real-time transportation management, just-in-time manufacturing logistics, and corporate decision-making. The range of potential application is expansive.

Fundamentally, any application that includes the following ideal domain properties may be a suitable candidate for BL:

- is currently taught to humans (or could be taught to humans)
- requires structured knowledge
- entails multiple representation shifts
- requires little domain knowledge to get started
- is not fundamentally reliant upon sophisticated perception (e.g., cyber domains)

APPENDIX C: DETAILS OF THE TECHNICAL APPROACH

Initial Elements of the Program

For the BLP to be successful, initial versions of a number of different elements need to be in place before the development of the full BL system can begin:

- A specification of the Interlingua
- A specification of the interaction language
- A fixed set of natural instruction methods, along with a formal specification of the contract for each NI method.
- A set of domains in which curriculum ladders will be built

Each of these will be discussed in this subsection.

Bootstrapped Learning Components and Interlingua

As mentioned above, because lessons build on previous lessons in BL, there must be a way to capture incrementally-learned knowledge. For this purpose each learning system will contain some number of *bootstrapped learning components* (“BLCs”), each of which captures knowledge of some aspect of a problem domain being learned. Since it is desirable for learning to occur at a conceptual level appropriate to the domain (e.g., learning how to manage a UAV might entail planning knowledge, but probably not the details of real-time flight control), learning is allowed to occur with respect to preexisting software that provides functionality useful in the domain. A typical BLC has four parts: predefined parameterized software that carries out actions or computes results in the domain, knowledge that forms a specification of how to use the predefined software and what it does, other initial knowledge relevant to this aspect of the domain (including an appropriate ontology), and learned knowledge.

Whereas the predefined software could take the form of almost any useful package, there is a need for a common language for representing the entire knowledge of a BLC. To create such a language in general is probably a Knowledge Representation–complete problem, and hence unfeasible. The BLP, however, requires something more tractable. In particular, it calls for a language that is simple enough that it could be **generated** and refined by learning processes, yet still expressive enough to support a range of interesting problems. This is referred to as the *Interlingua* and needs to support the representation of the following types of knowledge:

- *syntactic (ontological)*: domain objects and actions, function and predicate types including type restrictions on parameters and return values, etc.
- *logical*: world knowledge and inference rules
- *procedural*: knowledge of how to do things in the world
- *functional*: knowledge of how to compute complex functions by composing smaller ones.

Once the Interlingua has been defined, an appropriate set of initial BLCs can be provided as genetic knowledge. Because an initial version of the Interlingua will provide an important touch stone for the BL program, DARPA is actively developing a first implementation of the language prior to the start of the program. The current version of this language can be found by following this link in Appendix A. Example BLCs are also provided in the external materials as part of the three complete BL examples.

Interaction Language

Bootstrapped learning with a human instructor is an AI-complete problem. It would require solutions for the various modalities used to interact with the student such as natural language understanding, object recognition in computer vision, diagram understanding, etc. Developing robust solutions to modality-specific abstraction is not the focus of the BL program. Instead, for the purposes of this program, the interaction language is being developed to abstract the raw interactions that occur between humans during instruction. The goal of this language is not to maximize expressivity; rather, it is to provide a simplified representation capable of expressing materials communicated over a wide range of instructional methods for interesting subjects. By the same token, it must be simple enough that it is practical to use as a vehicle for learning. Ideally, the interaction language will be able to express 80% of the interactions found in formal and informal teaching today.

By employing a common interaction language, learning teams will be provided data from multiple domains in a clean, consistent format. The complex inputs from the real world will be abstracted in order to allow the learning system developers to concentrate their efforts on core learning processes. Due to the implications of the interaction language characteristics on learning process development, it makes sense for learning system developers to participate in the interaction language design.

Existing technologies are sufficient for an initial interaction language. Ongoing investigations have resulted in the identification of the following interaction modalities:

- *linguistic*: written and/or spoken signals to the electronic student
- *world perception*: the student's perception of the state of the world, or of a hypothetical state of the world (e.g., that an instructor would show as an example)
- *world action*: base actions in the world, i.e., actions that the student can take in the world when practicing, or that it can observe the instructor making in the world (e.g., in the NI methods *instruction by examples*)
- *gesture*: an instructor's physical gesture to make a point, e.g., pointing to features in other modalities, such as an action or object in the world
- *instructional cues*: specific ways of conveying the structuring of training materials, e.g., the dependencies between lessons, the objective for a given lesson, etc.

Many common methods of instruction used by people today can be built on this modest interaction language. For example, teaching:

- by feedback on student performance,

- from examples,
- by demonstration,
- by providing worked solutions,
- by feedback from world,
- by reasoning about failures, and
- by practicing.

The current specification of the interaction language is accessible via a URL provided in the external materials section in Appendix A. It is expected that the language will evolve somewhat over the life of the program.

Natural Instruction Methods

Natural instruction (NI) methods describe different ways of providing instruction. Figure 8 depicts the current set of identified NI methods as a tree with instruction types represented at the first level, labeled “Instruction From.” The investigation of human to human instruction undertaken by DARPA has resulted in the identification of four groups of NI methods. It is anticipated that all learning proposals will address each of these areas, though their characterization and division of all instruction into a set of methods may differ from the taxonomy provided herein. Additionally, teams may argue that other methods of Natural Instruction are critical, and should be included as well. The four principal types of instruction are:

- *Instruction by telling.* The instructor provides generalized statements that are descriptive in nature, e.g., linguistically. The instructor may also combine utterances with pointing or selection to the current state. Note that for BL, telling does not connote repeated study for memorization as it can in humans, but rather the appropriate integration of that memorized knowledge. In other words, one utterance is sufficient for generating an integrated memory trace. An example of “telling” might be to say (in a formal, constrained language) that the area of a rectangle is its length times its width.
- *Instruction by examples.* The instructor may select objects or actions in the environment and may highlight important features with gestures or annotations. Unlike instruction by telling, this instructional method is completely grounded and bound to specific objects known to the student, and is provided in terms of specific percepts or observable actions. An example might be to present the system with a room with a length of 10, and a width 5 whose area is 50, and to additionally gesture at the length, width, and area of the room, in order to signify that there is a relation to be found.
- *Instruction using feedback.* When the student provides an answer or solution to a provided problem, the instructor provides feedback that is directly tied to that answer. Both the student’s solution and the instructor’s feedback may be provided with more

or less detail. As the complexity of learning increases, the student benefits more and more from feedback on special cases or cases in which there are indirect interactions among objects or steps. Therefore, feedback is a common method to teach students about complex metrics such as evaluation functions for solution quality assessment. Practicing to maximize some metric, with or without an instructor present, is a form of feedback instruction. For example, the student might assert the area of the room described above is 500. One plausible form of feedback is to say “wrong.” More detailed feedback might take the form of, “wrong, since 5 times 10 does not equal 500.” Both forms of feedback would occur within a controlled language.

- *Instructor-guided discovery.* Traditional machine learning focuses on the problem of discovering a model of presented data. In the case of instructor-guided discovery, the BL system is also continuously attempting to model various aspects of the data presented to the system (for example, the valid syntactic relationships between new terms). The difference here is that the instructor has some understanding of what the student does and does not know. Thus, the instructor will structure materials in ways that minimize the difficulty of the discovery process. The student is expected to take advantage of this, by continuously searching for specific types of patterns from the instructor. For example, the student perceives coincidences in the data as intended by the instructor to lead to appropriate generalizations. Such an inference would not be made about random data. Even this indirect method is not focused on *discovery* of a new pattern; rather it is a method of *communicating* that pattern from instructor to student. Consider another example: imagine that the instructor provided many examples of $\langle X, Y \rangle$ pairs in teaching a concept, in this case the concept of “within 10 units of the Cartesian origin.” Imagine further that the instructor only chose points that were on the $X = Y$ diagonal line. Even though the instructor did not explicitly teach or mention that line, the student always seeks simplifying representations of the presented data as one of its indirect methods, and it learns a mapping of all input data onto a simplified representation of the data involving only one parameter. The student does not know why this line is important, but assumes it must be since the teacher is always presenting information in ways that lead it toward important concepts. In order to circumscribe the problem of learning by indirect methods we must assume that: (1) ‘simple’ discoveries are made, and (2) only discoveries that fall into some pre-enumerated classes of discovery types are made. Each ‘type’ of discovery constitutes a new NI-method within the category of Instructor-guided discovery.

Within each of these four NI method categories DARPA has investigated several possible specific NI-methods. The second level “Knowledge provided / generated” shown within the tree in Figure 8, depicts 12 such methods, in which knowledge is either transferred from the instructor or refined. Each of these approaches corresponds to exactly one NI method contract at the third level (“NI method contracts”). For example, one NI method is to have the instructor teach relations or functions by providing specific examples while gesturing at relevant features. In another NI method the teacher would provide feedback on student performance using an explanation (clearly, Feedback NI methods would necessitate an automated instructor).

The fourth level of the tree in Figure 8 depicts some of the learning processes (“Learning Algorithms”) in a learning system. In general, more than one learning process can be applied to a particular NI method.

Although each team is free to decompose the space of instruction differently, each team is expected to support at least these NI methods. DARPA recognizes that this articulation of the space is tentative and certainly incomplete. Nonetheless, it has been found to be adequate for supporting the instruction of a number of very general and challenging curricula. Teams may propose additional NI-methods though they may also simply focus on the methods we have identified as they may suffice for the BLP. If a team does propose additional NI-methods, it must explain how interactions could be encoded, and must argue for the ubiquity and centrality of the proposed methods.

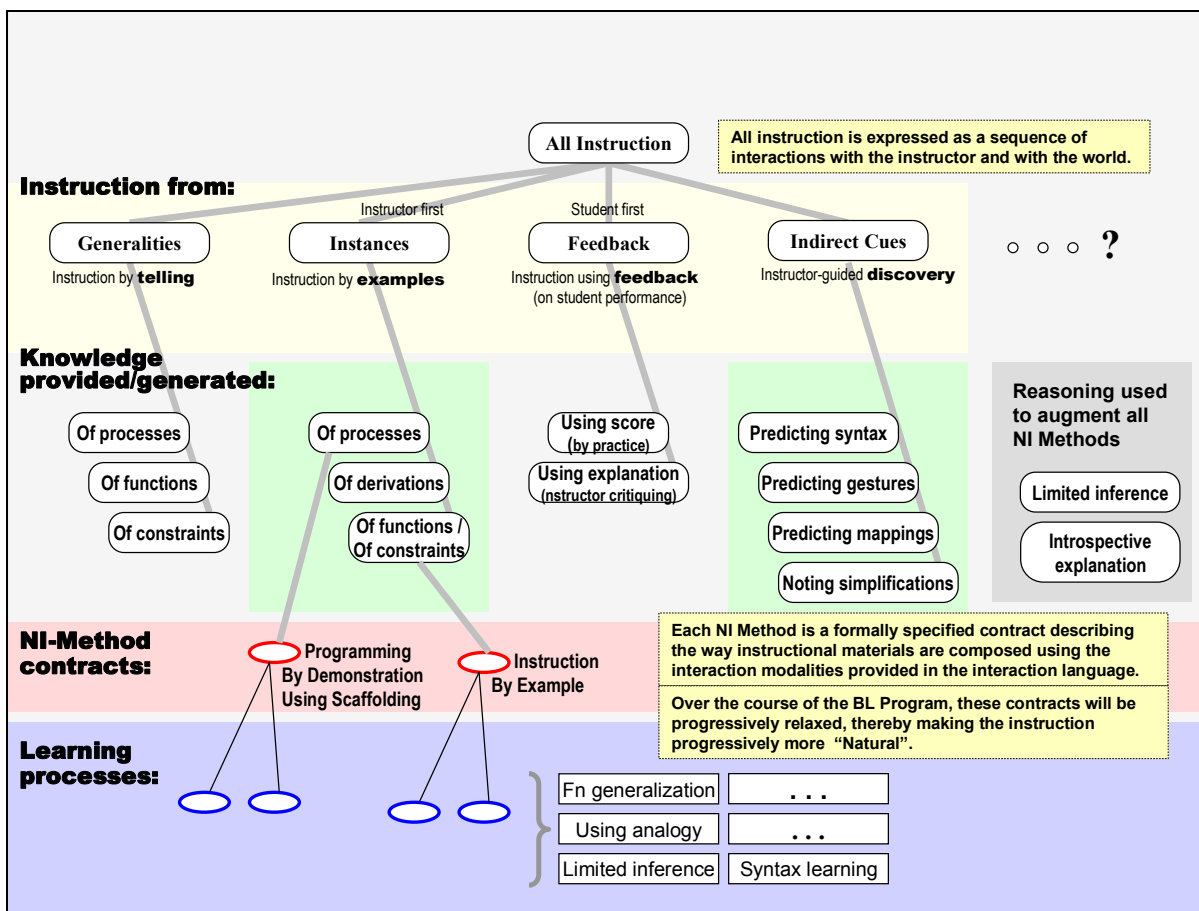


Figure 8: Taxonomy of Natural Instruction Methods

Each NI method is defined by an *NI method contract*. The NI method contract constitutes an agreement between the curriculum team and the learning team, defining for each NI method they have agreed to support, the specific interactions (see the layer titled “Learning processes” in Figure 8) that are allowed between any curriculum and any learning system. This contract is an important part of the BL program, as it provides a plain English

specification of the exact capabilities expected from the learning algorithms associated with each NI-method. Since the learning teams cannot control the data provided to their systems by the curriculum teams, this is their method of ensuring that they are building systems with sufficient generality for the scope of tasks to be addressed.

In order to provide some understanding of what an NI method contract might contain, an example is provided that conveys important aspects of a possible contract for learning from a demonstration:

The world action modality would be used to receive a stream of actions executed by the instructor. Furthermore, the world perception modality would be used to receive an interwoven stream of perceptual changes that occurred as a consequence of those actions. Instructor utterances and gestures could then be used to comment on the action that had just occurred, which could also be interwoven in the stream of events.

In this NI method, the instructor can utter the following formal phrases:

- RelevantNow(<gesture-target>)
- ReasonForUpcomingBranch(<gesture-targets>)
- MyCurrentGoalIs(<GoalName>)
- IsSubgoalOf(<GoalName1>, <GoalName2>)

The NI-Method contract would go on to explain in English how ‘IsSubGoalOf’ defines a DAG of goal tokens, and that the instructor can group actions hierarchically by stating their current goal. ‘Relevant-Now’ and ‘ReasonForUpcomingBranch’ are utterances that provide hints about how perceptual features relate to aspects of the procedure to be induced. GestureTargets are the objects or features of objects that were “pointed at” by the instructor when the utterance was made.

It is ideal for learning algorithms to accept these rich hints, but it is important that they be developed in a way that is robust to the inevitable missing or noisy hints that a real human would provide.

There are several considerations outlined below, which will be used to determine the appropriateness of each NI-method contract. A consensus from all learning teams and the curriculum team will be required for the ratification of these contracts. DARPA will be the final arbiter for determining the appropriateness of each, though the decision will be driven by the constraints associated with the learning teams’ algorithms. Thus, some contracts will not be viable simply because corresponding learning algorithms are not feasible.

Prospective learning system developers should propose their own set of NI methods and learning processes, in which at least one learning process is applicable to each NI method.

Desirable characteristics of an NI method for the BL program include the following:

- *Naturalness*: The NI method is an abstraction of a plausible interaction between human instructor and student.

- *Effectiveness*: The NI method is much simpler than any other method for instructing the same material (e.g., easier than directly programming the same task).
- *Instructional efficiency*: The NI method is sensitive to the implied instructor time needed for its application.
- *Applicability*: The NI method can be used extensively across a diverse range of domains.
- *Encodability*: The NI method's interactions can be encoded into a curriculum ladder in the interaction language with relative ease.
- *Computational efficiency*: A computationally efficient learning process can be developed to learn from the interactions employed by this NI method.
- *Robustness*: It is feasible for learning processes to handle missing data or noisy inputs, as well as “haphazard” instruction, using this NI method.
- *Encapsulatability*: A central aim of the BLP research agenda is to provide a test bed where new learning techniques can be rapidly modified and retested. Such a test bed is critical to support the inherently empirical nature of this investigation. Thus, a suitable NI method must be functional in an automated instructional context. For example, one might propose using an NI method involving instructor feedback on student solutions. In this case, one would need to characterize and limit the cost of providing an “electronic instructor” capable of delivering the class of relevant interactions with the needed feedback and without the benefit of human intervention.

Ni method contracts are written using the interaction language and interaction modalities. Part of the information provided is what the educational psychology community refers to as *scaffolding*. These are not facts about the world, but rather guidance in the form of hints.

Scaffolding – knowledge expressed in some interaction modalities of the interaction language which convey guidance or hints from the instructor, rather than knowledge which conveys facts from the instructor or world.

The term scaffolding is used herein to refer to content in NI method contracts that is hint information.

In order to build a complete learning system quickly, and then refine it over the lifetime of the BL program, it is useful to define “relaxation trajectories” for each of the NI method contracts. Early in the program it will be convenient for the contracts to provide simplifying information which is not plausibly part of a natural instructional interaction. For example, a contract could stipulate that the instructor always mentions every predicate needed for a new concept before attempting to teach it by example. While human instructors might provide such scaffolding, it is implausible that they would always be so thorough. Nonetheless, one possible relaxation trajectory is to begin with that scaffolding, but then to relax the amount and clarity of scaffolding as the program progresses. Another example involves stipulating that training for each NI method occurs separately. The associated relaxation might be to

allow the interactions from multiple instructional methods to be interspersed without any delineation provided, which happens often in actual human instruction.

Relaxation trajectory – a program policy that specifies progressive leniency in NI method contracts over the course of the BL program in order to move toward more natural learning.

Each team is expected to provide innovative ideas about how general these contracts might be, and what relaxation trajectories it would accept. Weight will be given to proposals that can articulate convincingly plausible approaches to very general NI-method contracts and aggressive relaxation trajectories.

Examples of classes of relaxation trajectories considered by DARPA include:

- Allowing method data to be mixed freely
 - Initially indicate which NI method applies to each segment
 - Later, no longer specific which method applies
- Allowing missing and erroneous scaffolding indicators with increasing frequency
- Removing entire classes of scaffolding in some lessons, particularly if providing scaffolding is expensive
- Fading – in the course of a single lesson, provide ample scaffolding for early examples and then reduce scaffolding for later examples

The ultimate objective is for the instruction to be as close as possible to human instruction by the end of the program and for the NI methods to support general purpose instructable computing.

Additional commentary on NI methods can be found in the external materials, referenced in Appendix A. It should be noted that not all the materials discussing NI-methods, including the example scenarios that use NI-methods, were developed in an attempt to gain a better understanding of such methods. In many cases this development work occurred at the direction of DARPA, but was performed by an external organization. Thus, these examples should not be taken as definitive, but rather as possible interpretations and directions one might follow in realizing a BL system. DARPA is not committed to specific interpretations at this point, and is open to proposals with alternative characterizations.

Ancillary Competencies

Each NI-method contract specifies a constrained way that instructional information should be provided. This is necessary since we must have some basis for the algorithm development work. At the same time, there are common ways that the instructional material could “almost” contain all of the right information, yet still not conform to the NI-method contracts, and thus not allow bootstrap learning.

Several “ancillary competencies” suggest themselves as broad remedies for certain types of such mismatches. Two of these are described as examples of such ancillary processes, and because DARPA believes both are important parts of a full BL solution. It is expected that there may be other important ancillary processes, thus, innovative thinking about this class of capability is encouraged within the proposal.

- *Limited Inference.* One could imagine the learning by example NI-method contract might stipulate that the value of the target feature for each example be explicitly provided in the instructional material. After all how can you learn from examples if you don’t know the designated value in each example? But imagine that the value of this feature was not explicit, but was easily inferred using a piece of existing knowledge. A human student would have no problem doing that inference, and thus be able to learn by example even when the values were not explicitly asserted by the instructor. Judicious use of limited inference by the integration architecture could be used in myriad ways to transform the instructional input into a form that conforms to the NI method contracts.
- *Introspective Methods.* Another common transformation would be to employ a model of one’s own behavior and a model of one’s own reasoning in order to connect instructional percepts to terms and concepts that are internal to the BL system’s BLCs. As an example, imagine the learning agent is executing a learned process in the world, and some negative consequence occurs. It turns out the definition that agent has for some predicate is incorrect. As a consequence, an incorrect branch on a conditional in the currently-executing procedure was mistakenly taken. As a consequence of that, an incorrect action was taken, which resulted in ill effects in the world. Recognizing this situation as a possible learning opportunity for that incorrect predicate would require introspective reasoning over the knowledge that was in the BL system, and the interaction of that knowledge with the execution system and with the world. Providing such an ancillary capability for use by the integration architecture is important since it greatly expands the situations where instructional experience can be transformed into NI method contracts. At the same time such capabilities are far from understood today. Thus an important area of innovative thinking in each learning team proposals will deal with this and possible other ancillary capabilities employed by the integration architecture.

Curricula and Curriculum Ladders

All of the interactions between the instructor or the world and the electronic student required to teach a single curriculum are encapsulated in a single *curriculum ladder*, which is a chronological series (potentially represented by a graph) of lessons. In principle, any task domain could be encoded as a curriculum. Here we discuss characteristics of tasks that would lend themselves to being encoded as BL curricula:

- They seek to impart “natural” competencies, i.e., tasks currently taught or that could be taught to humans
- They provide complex instruction: exhibit a measurable, practical increment in human performance following instruction, provide multiple layers of concepts and

sub-concepts and/or procedures and sub-procedures, and require relational knowledge and representation shifts

- They might reside in cyber domains where the perception problem is easier to sidestep
- They correspond to curricula that are inexpensive to build: they require limited background knowledge, leverages existing simulators and training materials, and are easily expressed as content suitable for human consumption (especially for the domain that is chosen for the human comparison experiments).

Each curriculum will be packaged to include world simulators, relevant background knowledge, and a structured tree of problem generators. Since each curriculum ladder is a self-contained, complete testing environment, it will be easy to develop rapidly and test new learning processes. The BLP aspires to provide datasets that will drive research on instructable learning in much the same way that the Irvine Repository drove supervised induction in the 1980s.

Fairly complete examples of curricula have been produced for several domains: the blocks world, an architectural CAD application, and managing a UAV. The details of these examples can be found in the external materials, which are referenced in Appendix A.

Learning Processes

In bootstrapped learning, the fundamental nature of machine learning is changed with respect to traditional ML. The requirement on a BL procedure is not to use tons of data to discover structure within a domain, but rather to capture and appropriately integrate the structure that is communicated more directly by way of instruction.

In order for a BL learning system to be compatible with an instructional approach, it must possess at least one learning process for each type of instructor-student interaction possible, i.e., for each NI method contract.

In the process of assimilating an entire curriculum, each BL lesson, or rung of the ladder curriculum, should result in some part of a BLC being developed via updates to relevant Interlingua content. For indefinite bootstrapping to be possible, it will be necessary to have learning processes that cover all four major knowledge categories of the Interlingua: syntactic (ontological), logical, procedural, and functional, as defined previously. Ideally, a learning process would exist for each combination of NI method and Interlingua knowledge type. However, this likely exceeds the practical constraints of the BLP.

Examples of possible learning processes:

- A *syntax-learning* learning process would monitor input modalities, trying to infer new terms, new relationships, and their arguments. For example, it might learn that “rear” applies to physical objects and returns a 3-D location.
- An *annotated-examples* learning process would only be invoked when the instructor uses an appropriate *instruction by examples* NI method. This process might be based

on existing example-based induction techniques, but the incorporation of gestures and linguistic hints would both add powerful constraints.

- A *by-refinement* learning process would accept feedback on performance and use that to guide an update process. Theory refinement, within the ML community, is one example of such a learning process.
- A *by-watching* learning process would receive a stream of instructor comments and gestures (e.g., about relevant features, asserting the current goal, etc.), along with actions from the instructor, and perceived changes in the world. Using this input the learning system would induce the procedure being executed by the instructor. This is related to the Programming by Demonstration work that has been done within the AI community.

Other existing learning techniques that might be incorporated into a learning system include function generalization and analogical learning.

As mentioned earlier, each step of incremental bootstrapped learning results in a modification to a BLC, which takes the form of the creation or modification of one or more Interlingua statements. It is apparent that there are a few critical types of BLC transformations, or “shifts” that result from one or more such modifications to a BLC. Although this list is incomplete, the following are types of shifts that have been identified:

- *Compositional shift.* Knowledge is incrementally built up, based on the existing knowledge in a BLC.
- *Instantiational shift.* A BLC is selected, one or more parameters of the existing software substrate are selected, and then they are instantiated with appropriate values.
- *Transformational shift.* A new way is discovered to use an old BLC. This type of shift is more difficult to learn, but is important to successful BL. Analogical reasoning would fit in this category.
- *Representational shift.* A mapping is found from one representation of knowledge to another. The impetus here is that the original representation space is somehow not appropriate for the necessary reasoning/performance, and so (under direction of the instructor) the BL system learns a new representational space. This new representational space takes the form of a BLC with a new syntax and induces the mapping to and from the original representation space. Thus, future performance benefits from the new space, and Bootstrap Learning can build from it.

Other ideas for learning processes are provided in the worked-out examples available on the BLP website (see Appendix A).